Information Builders.

WebFOCUS

Summary of New Features
Version 5 Release 2

DN4500463.0304

EDA, EDA/SQL, FIDEL, FOCCALC, FOCUS, FOCUS Fusion, FOCUS Vision, Hospital-Trac, Information Builders, the Information Builders logo, Parlay, PC/FOCUS, SmartMart, SmartMode, SNAPpack, TableTalk, WALDO, Web390, WebFOCUS and WorldMART are registered trademarks, and iWay and iWay Software are trademarks of Information Builders, Inc.

Due to the nature of this material, this document refers to numerous hardware and software products by their trademarks. In most, if not all cases, these designations are claimed as trademarks or registered trademarks by their respective companies. It is not this publisher's intent to use any of these names generically. The reader is therefore cautioned to investigate all claimed trademark rights before using any of these names other than to refer to the product described.

Copyright © 2004, by Information Builders, Inc and iWay Software. All rights reserved. Patent Pending. This manual, or parts thereof, may not be reproduced in any form without the written permission of Information Builders, Inc.

Preface

This manual describes the new features available in WebFOCUS Version 5 Release 2. It is intended for all levels of users, including application developers, administrators, and end users.

How This Manual Is Organized

This manual includes the following chapters:

Chapter/Appendix		Contents
1	WebFOCUS Reporting Language Enhancements	Describes new features and enhancements to the WebFOCUS Reporting Language.
2	Developer Studio Enhancements	Describes new features and enhancements to Developer Studio.
3	Managed Reporting Enhancements	Describes new features that enhance Managed Reporting, including User Administration Services, Dashboard, the applet environment, and migration and extract utilities.
4	Ad Hoc Reporting Enhancements	Describes new features that enhance ad hoc reporting capabilities. These enhancements include OLAP, Report Assistant, and Graph Assistant features.
5	ReportCaster Enhancements	Describes new features and enhancements to ReportCaster.
6	WebFOCUS Client Enhancements	Describes new features that enhance the WebFOCUS Client and National Language Support (NLS).
7	WebFOCUS Reporting Server Enhancements	Describes new features that enhance the WebFOCUS Reporting Server.
8	Data Adapter Enhancements	Describes support for new DBMSs, new versions of existing DBMSs, and features within DBMSs.
9	WebFOCUS Maintain Enhancements	Describes the enhancements that have been made to the Maintain Language, Server, and Development environments.

Chapter/Appendix		Contents
10	ETL Manager Enhancements	Describes enhancements to ETL Manager.
11	Resource Analyzer and Resource Governor Enhancements	Describes enhancements to Resource Analyzer and Resource Governor.
A	New Feature Roadmap for WebFOCUS Version 5 Release 2.3	Provides a listing of the new features that are specific to WebFOCUS Version 5 Release 2.3.

Documentation Conventions

The following conventions apply throughout this manual:

Convention	Description
`THIS TYPEFACE` or `this typeface`	Denotes syntax that you must enter exactly as shown.
`this typeface`	Represents a placeholder (or variable) in syntax for a value that you or the system must supply.
underscore	Indicates a default setting.
this typeface	Represents a placeholder (or variable), a cross-reference, or an important term. It may also indicate a button, menu item, or dialog box option you can click or select.
this typeface	Highlights a file name or command.
Key + Key	Indicates keys that you must press simultaneously.
`{ }`	Indicates two or three choices; type one of them, not the braces.
`[ ]`	Indicates a group of optional parameters. None are required, but you may select one of them. Type only the parameter in the brackets, not the brackets.
`|`	Separates mutually exclusive choices in syntax. Type one of them, not the symbol.

Convention	Description
. . .	Indicates that you can enter a parameter multiple times. Type only the parameter, not the ellipsis points (…).
. . .	Indicates that there are (or could be) intervening or additional commands.

Related Publications

To view a current listing of our publications and to place an order, visit our World Wide Web site, http://www.informationbuilders.com. You can also contact the Publications Order Department at (800) 969-4636.

Customer Support

Do you have questions about WebFOCUS?

Call Information Builders Customer Support Services (CSS) at (800) 736-6130 or (212) 736-6130. Customer Support Consultants are available Monday through Friday between 8:00 A.M. and 8:00 P.M. EST to address all your WebFOCUS questions. Information Builders consultants can also give you general guidance regarding product capabilities and documentation. Please be ready to provide your six-digit site code (*xxxx.xx*) when you call.

You can also access support services electronically, 24 hours a day, with InfoResponse Online. InfoResponse Online is accessible through our World Wide Web site, http://www.informationbuilders.com. It connects you to the tracking system and known-problem database at the Information Builders support center. Registered users can open, update, and view the status of cases in the tracking system and read descriptions of reported software issues. New users can register immediately for this service. The technical support section of www.informationbuilders.com also provides usage techniques, diagnostic tips, and answers to frequently asked questions.

To learn about the full range of available support services, ask your Information Builders representative about InfoResponse Online, or call (800) 969-INFO.

Information You Should Have

To help our consultants answer your questions most effectively, please provide the following information when you call:

- Your six-digit site code (*xxxx.xx*).
- Your WebFOCUS configuration:
  - The front-end you are using, including vendor and release.
  - The communications protocol (for example, TCP/IP or HLLAPI), including vendor and release.
  - The software release.
  - Your server version and release. You can find this information using the *Version* option in the Web Console.
- The stored procedure (preferably with line numbers) or FOCUS commands being used in server access.
- The name of the Master File and Access File.
- The exact nature of the problem:
  - Are the results or the format incorrect? Are the text or calculations missing or misplaced?
  - The error message and return code, if applicable.
  - Is this related to any other problem?
- Has the procedure or query ever worked in its present form? Has it been changed recently? How often does the problem occur?
- What release of the operating system are you using? Has it, WebFOCUS, your security system, communications protocol, or front-end software changed?
- Is this problem reproducible? If so, how?
- Have you tried to reproduce your problem in the simplest form possible? For example, if you are having problems joining two data sources, have you tried executing a query containing the code to access a single data source?
- Do you have a trace file?
- How is the problem affecting your business? Is it halting development or production? Do you just have questions about functionality or documentation?

User Feedback

In an effort to produce effective documentation, the Documentation Services staff welcomes your opinions regarding this manual. Please use the Reader Comments form at the end of this manual to relay suggestions for improving the publication or to alert us to corrections. You can also use the Documentation Feedback form on our Web site, http://www.informationbuilders.com.

Thank you, in advance, for your comments.

Information Builders Consulting and Training

Interested in training? Information Builders Education Department offers a wide variety of training courses for this and other Information Builders products.

For information on course descriptions, locations, and dates, or to register for classes, visit our World Wide Web site (http://www.informationbuilders.com) or call (800) 969-INFO to speak to an Education Representative.

Contents

1. WebFOCUS Reporting Language Enhancements 1-1

Calculating Trends and Predicting Values 1-2
- Calculating Trends and Predicting Values With FORECAST 1-3
- FORECAST Processing 1-4
- Using a Simple Moving Average 1-6
- Using Single Exponential Smoothing 1-9
- Using Double Exponential Smoothing 1-12
- Using Triple Exponential Smoothing 1-13
- Using a Linear Regression Equation 1-16
- Calculating Trends and Predicting Values Using REGRESS 1-18

Calculating Cumulative Normal Distribution 1-21
Grouping Numeric Data Into Tiles 1-25
Grouping Sort Field Values in Ranges 1-27
Indicating Total Number of Pages 1-29
Manipulating Summary Values 1-31
- Controlling Summary Line Processing 1-37
- Using Prefix Operators With Calculated Values 1-38
- Using Multiple SUB-TOTAL or SUMMARIZE Commands 1-40

Handling Missing Data in HOLD Files 1-43
- Controlling Missing Values Propagated to a HOLD File 1-43
- Holding Missing Values 1-47
- Propagating Missing Values to a Variable Length Delimited HOLD File 1-48

Using Variable-Length Character Expressions 1-50
- Using A*n*V Fields With Functions 1-52
- LENV: Returning the Length of an Alphanumeric Field 1-53
- LOCASV: Creating a Variable Length Lowercase String 1-54
- POSITV: Finding the Beginning of a Variable Length Substring 1-56
- SUBSTV: Extracting a Variable Length Substring 1-59
- TRIMV: Removing Characters From a String 1-61
- UPCASV: Creating a Variable Length Uppercase String 1-63

Validating Function Arguments 1-66
Financial Reporting Language Enhancements 1-70
- Defining an FML Hierarchy 1-71
- Displaying a Dynamic FML Hierarchy 1-73
- Consolidating Data in an FML Hierarchy 1-78
- Indenting Row Titles in an FML Hierarchy 1-85
- Returning FOR Field Values in a Computed Column 1-87
- Formatting an FML Report 1-89

Graph Enhancements 1-93
Saving a Graph as a GIF File 1-93
Altering X- and Y-Axis Values Using Linear Regression 1-97
New Graph Type Options 1-97
Sending Graph Output Directly to a Printer 1-97
Expanded Limits 1-98
Increased Number of Display Fields 1-98
Increased Number of Columns Generated by ACROSS 1-98
Increased Numbers of Headings and Footings 1-98
Increased Space for Column Titles 1-98
Prompting for Variables in CGI/Servlet 1-99
Including Script Files in Reports 1-104
Specifying Holidays 1-104
Establishing a Non-Standard Alias 1-105
Compiling Expressions Containing DEFINEs 1-105
Exiting on Error 1-106
Converting Case for File Names and Directories 1-107
Adding Borders for Emphasis 1-107
Aligning Decimals in a Multi-Line Heading 1-111
Using Fields to Define Formats 1-116
Punctuating Numbers 1-118
Displaying a Leading Zero in Decimal Numbers 1-119
Selecting a Currency Symbol for Display 1-121
Displaying a Minus Sign After Negative Data 1-124
Displaying Compound Reports in PS Format 1-125
Excel Integration Enhancements 1-129
Selecting a Language for Excel Requests 1-137
Setting the Default Column Width for Excel 1-138
Stacking Columns With Multiple FOLD-LINEs 1-138
Specifying Multiple Drill-Down Links 1-140
Navigating Reports From a Table of Contents 1-141
Grouping Sort Fields for Display 1-147
Producing Row Totals for ACROSS Field Values 1-149
Output Formats for Application Processing 1-151
COM Format 1-152
COMT Format 1-153
TAB Format 1-153
TABT Format 1-154
Adjusting Field Displays 1-154
Reducing Multiple Spaces in a Field to a Single Space 1-155
Removing a Character From a Field 1-156
Removing Leading or Trailing Occurrences of a Pattern 1-158

PostScript Type 1 Fonts for PS and PDF Reports 1-160
How Type 1 Fonts Are Used 1-160
Controlling the Paper Source for a PS Printer 1-161
SET Parameter Summary 1-162
Functions Summary 1-165

2. Developer Studio Enhancements 2-1

New Architecture 2-2
Managed Reporting Developer for Windows 2-3
Enhanced Explorer 2-4
Project Wizard 2-4
Redesigned Deploy Wizard 2-5
Dimensions Tool 2-6
Synonym Enhancements 2-7
Date Format and Time Display Options 2-8
Comma Suppression Display Option 2-9
Percent Edit Display Option 2-10
Report Viewing in a Browser 2-10
PS Format in Compound Reports 2-11
Excel Enhancements 2-11
Comma-delimited and Tab-delimited Output Formats 2-12
Configuration for ANSI (Windows) Format 2-13
Creating and Styling Reports With the Report Painter 2-14
Summary of Report Painter New Features 2-14
Creating Multiple Drill Downs With the Report Painter 2-21
Calculating Trends and Predicting Values in Developer Studio 2-26
Calculating Trends and Predicting Values With FORECAST in Developer Studio 2-26
Designing a User Interface With the Resource Layout Painter 2-35
Form Controls With User-supplied Values 2-35
Enhanced Customization of Web Objects 2-35
Setting Static Parameter Values 2-36
Updating Parameters in the Resource Layout Painter With Report Painter Changes 2-36
Automatically Deleting a Form Control Associated With a Parameter 2-39
Filtering Dynamic Parameter Values 2-40
Adding an Image With a Hyperlink to the Layout 2-43
Displaying Radio Buttons and Check Box Lists in Multiple Columns 2-44
Specifying Different Names for Dynamic Parameter Values 2-46
Hiding Borders Around Frames 2-47
Applying Parameter Values to External Reports 2-47
Source File Management 2-48

Update Assist 2-49
Dynamic List Generation 2-49
Results Page 2-50
Enhanced DBA Prompting 2-50
Enhanced Data Retrieval Error Messages 2-50
Tree Navigation Enhancements 2-51
Enhanced Date Field Validation 2-51
Validation of User-Supplied Data in Update Assist Applications 2-51
Choosing a Color Scheme for an Update Assist Application 2-52
Customizing Update Assist Designs 2-52
Calendar Control for Date-Formatted Fields in Update Assist 2-53
Right-Click Menu Options on Trees 2-53
Sorting Tables 2-54
Financial Report Painter Enhancements 2-54
Reporting Dynamically From a Hierarchy 2-55
Indenting Row Titles 2-56
Displaying an FML Hierarchy With the Financial Report Painter 2-56
Consolidating Data in a Hierarchy in Developer Studio 2-62
Formatting Enhancements 2-64
SET Parameters 2-69
Functions for Displaying Fields 2-70

3. Managed Reporting Enhancements 3-1

User Administrator 3-2
Developer User Type 3-2
Managed Reporting Developer for Windows 3-3
Change Management 3-4
Security 3-4
Servlet Exit 3-5
Integrating Managed Reporting Security With Basic Web Server Authentication 3-5
LDAP Authentication 3-6
Updated Presentation 3-6
Reporting Object Component Validation 3-7
New Create Synonym Behavior 3-7
Multiple Servers and Application Path 3-7
Custom Reports in Managed Reporting 3-8
Execution of a Custom Report Using -INCLUDE 3-11
Deferred Receipt 3-12
Migration Utility 3-12
Repository Extract Utility 3-12
Creating Multiple Content Pages 3-13
Creating a Group View 3-13
Unique Customizations for Public Views 3-14

Removing the Banner 3-14
Customizing Content Block Layouts 3-14
Saving Most Recently Used Reports 3-15
Creating a My Favorite Reports List 3-15
Creating Custom Toolbars 3-15
Stand-Alone Domain/Role Trees 3-17
Optionally Displaying Frame Separators 3-18
Hiding the Display of Content Blocks 3-18
Default Public User 3-18
Domain Searches 3-18
User Options Settings 3-19
Custom Reports in Dashboard 3-19
Filters For Standard Reports in Dashboard 3-20
Content Pages: ReportCaster and Report Library 3-20
Dynamic Tracing 3-20

4. Ad Hoc Reporting Enhancements 4-1

OLAP 4-2
Drag-and-Drop Dimensions and Measures 4-2
Right-Click in a Report 4-3
Hidden Reports 4-5
Tiles 4-6
HTML Report Assistant 4-8
Report Presentation 4-9
Creating New Columns 4-10
Selecting Records for a Report 4-11
Designing Joins 4-15
End User Drill Down Capability 4-16
Performing OLAP Operations Directly From the Report 4-16
HTML Graph Assistant 4-17
Addition of Several Templates to the Styles Tab 4-18
Updated Styles Tab to Include Graph Types and Graph Styles 4-18
Addition of the Preview Screen 4-18
Auto Update Feature Added to the Preview Screen 4-19
HTML Report Assistant and HTML Graph Assistant 4-19
DBA and DBAFILE Support 4-19
Creating a WHERE Statement Using Date and Time Values 4-20

5. ReportCaster Enhancements 5-1
Architecture 5-2
Migration Tools 5-2
Configuring the ReportCaster Web Application 5-2
Configuration Tools 5-3
Security Configurations 5-3
ReportCaster Authentication Exit 5-4
ReportCaster User Credentials 5-4
Configuring Access to the User Interfaces 5-4
Development and Administration Interface 5-5
ReportCaster Server Configuration Tool 5-5
Creating and Managing ReportCaster User IDs 5-5
Distribution Enhancements 5-6
Distributing a Task 5-6
Distribution Enhancements When Creating a Schedule 5-7
Scheduling Wizard and HTML User Interface 5-8
Distribution Enhancements When Using the Scheduling Wizard 5-8
Report Library 5-9
ReportCaster Console 5-10
Log Enhancements 5-10
Configuring ReportCaster for NLS 5-11
ReportCaster API 5-17
Two-Way Email API 5-18

6. WebFOCUS Client Enhancements 6-1
Introducing the WebFOCUS Client Console 6-2
Accessing the WebFOCUS Client Console 6-2
Configuring WebFOCUS Reporting Server Settings 6-4
Adding or Changing Server Settings 6-4
Configuring Deferred Server Mappings 6-7
Configuring Special Services 6-9
Configuring WebFOCUS Client Settings 6-10
Configuring WebFOCUS Startup Parameters 6-10
Changing WebFOCUS Client Settings 6-15
Redirection Settings 6-18
Customizing WebFOCUS Client Settings 6-20
NLS Settings 6-20
WebFOCUS Client Tracing 6-22
Quick Links 6-22
National Language Support for International Computing 6-23

Using the Dynamic Language Switch 6-24
Dynamic Language Switch Feature 6-25
Directory Structures 6-28
Editing Language Values 6-29
Customizing the User Interface 6-30
Architecture 6-31
Locating the Name of a Resource File 6-32
Interpreting the Resource File 6-33
Editing the .htm File 6-33

7. WebFOCUS Reporting Server Enhancements 7-1

Installing WebFOCUS Reporting Server Remotely 7-2
Controlling WebFOCUS Reporting Server Remotely 7-2
Agent Services 7-2
Blocking New Connections on a Running Server 7-4
Enhanced Connection Management 7-4
Specifying the Location of User Profiles 7-4
Configuring National Language Support 7-5
Step 1. Determine Which Code Page You Need 7-6
Step 2. Configure the WebFOCUS Reporting Server for NLS 7-9
Step 3. Configure the WebFOCUS Client for NLS 7-12
Configuring the Server and Client for Baltic Code Pages 7-15
Specifying the Number of Lines in the EDAPRINT Log 7-16
Automatic Self Recovery 7-17
Tree View for Procedures 7-17
Text Console Enhancements 7-17
Communicating With a FOCUS Database Server 7-18
Windows Authentication Methods 7-21
Assigning Administrative Privileges 7-22
Protecting the WebFOCUS Reporting Server Console 7-23
Windows Trusted Client 7-23
HTTP Basic Authentication 7-24
SSL/TLS Support 7-24
Lawson Security Adapter 7-25
C2 Security Support 7-25
Data Adapter Support 7-26
Creating Synonyms 7-26
Accessing Files on z/OS and OS/390 7-27
External Sort 7-28
Hiper Memory 7-28
Accessing z/OS and OS/390 Applications From the WebFOCUS Reporting Server Console 7-29
Accessing z/OS and OS/390 VSAM Files Using XMI Server 7-29
Security Types 7-31

Server Accounting Using SMF Records 7-31
Workload Manager Enclave Integration 7-32
Support for Additional Operating Systems and Releases 7-32
Support for Large Files on HP and Solaris 7-32
Displaying Listener Statistics in the WebFOCUS Reporting Server Console 7-33
MQ XML Listener 7-33
SOAP XML Listener 7-33

8. Data Adapter Enhancements 8-1

Data Adapter for Adabas 8-2
Setting Connection Attributes 8-2
Adabas Transparent Superdescriptors 8-4
Adabas Hyper and Phonetic Descriptors 8-5
Adabas ORDER Fields in the Indexed Field List 8-6
Adabas Internal Sequence Numbers (ISN) and Global Format Buffer IDs (GFBID) 8-6
Switching Access Mode Using NOPACCESS 8-8
Data Adapter for Allbase on HP-UX 8-9
Data Adapter for Micro Focus C-ISAM 8-9
Data Adapter for DBFILE 8-10
DBFILE Metadata 8-10
Data Adapter for DB2 8-10
Declaring DB2/CAF Connection Attributes From the WebFOCUS Reporting Server Console 8-11
DB2 FETCH FIRST *n* ROWS 8-12
DB2 Table Types on OS/400 8-13
Support for DB2 8.1 8-13
Data Adapter for Enterprise Java Beans 8-13
Data Adapter for Essbase 8-14
Describing the Measures Dimension in the Master File 8-15
Summing on Non-Aggregated Fields 8-15
Preventing Aggregation of Non-Consolidating Members 8-16
Suppressing Shared Members 8-17
Suppressing Zero Values 8-17
Suppressing Missing Data 8-17
Suppressing Zero Values and Missing Data 8-18
Support for User-Defined Attributes 8-18
Data Adapter for Informix 8-19
Data Adapter for J.D. Edwards 8-19
Data Adapter for Millennium 8-19
Data Adapter for MUMPS 8-19
Data Adapter for PeopleSoft 8-20
Data Adapter for Progress 8-20
Data Adapter for SAP BW 8-21

Data Adapter for SAP R/3 8-21
Security Support for the SAP R/3 Native Interface 8-22
BAPI Support in the SAP R/3 Interface 8-22
Join Support in the SAP R/3 Interface 8-23
Aggregation Support in the SAP R/3 Interface 8-23
IDoc Listener 8-23
Data Adapter for Siebel 8-24
Data Adapter for UniVerse on HP-UX 8-24
Data Adapter for VSAM 8-25
Java Applications Adapter 8-26
Writing a Java Class 8-27
Java Class Communications 8-27
Compiling and Running a Java Program 8-27
DATASET Attribute for a Segment in a Master File 8-28
DATASET Behavior in FOCUS Data Sources 8-29
COBOL FD Translator Enhancements 8-32
Database Selection From the Console 8-33
Field Identification as Read-Only 8-34
LIBPATH Display for DBMS 8-35
Long Qualified Field Names 8-36
Size Restrictions for Generated Alpha Fields 8-38
SQL Join Optimization 8-38
Text Fields for Relational Data Adapters 8-39
32K Text Strings for Text Fields 8-39
WHERE Optimization Clause for Text Fields 8-40
SUBSTRING Function Behavior 8-42
VARCHAR Support 8-42

9. WebFOCUS Maintain Enhancements 9-1
Tighter Integration With Developer Studio 9-2
Support for Standard WebFOCUS CGI Aliases 9-3
Access to the Installation Verification Program 9-3
Performance Enhancements 9-4
Deploying Only Changed Files With Smart Deploy 9-4
Running Components From the Developer Launch Console 9-6
Editing Master Files as Text 9-8
Customizing Parsing Activity 9-8
Searching for Text in a Project 9-9
Creating Applications With Update Assist 9-10
Applying a Cascading Style Sheet to a Form 9-10
Importing a Form Into a Procedure 9-12
Dynamically Managing the Display of Controls 9-13
Preselecting List Items by Value 9-13
New Sample Launch Page 9-14
Multiple-Level Tracing With Statement Trace 9-15
Application-Level Tracing With TYPE ON EDAPRINT 9-16
Retrieving WebFOCUS CGI Parameters 9-16
Importing and Binding Variables From a WebFOCUS Report 9-17
Changing WebFOCUS Settings From a Procedure 9-20
Issuing SQL Commands in a Procedure 9-30
Examining RDBMS Return Codes in a Procedure 9-32
WebFOCUS Maintain Functions 9-33
CHAR2INT 9-33
INT2CHAR 9-34
NLSCHR 9-34
GetAppCGIValue 9-35
Enhanced Screening Options 9-35
CONTAINS and OMITS 9-35
NOT_IN 9-35
EQ_MASK and NE_MASK 9-35
File Designations for Procedures 9-36
Testing Procedures With Run Procedure 9-37
Syntax Parsing in Scripts 9-37
Specifying Additional Search Paths 9-38
Support for Data Source Access Passwords 9-39
Coding Auto-Increment Columns 9-40
Concurrent FOCUS Data Source Access 9-42
Variable Varchar Data Type Support 9-43
Additional Microsoft Access Support 9-43
WebFOCUS Connector for Excel 9-44

10. ETL Manager Enhancements 10-1
Request Generation 10-2
Request Categories and Application Directories 10-2
Request Logs 10-3
Request and Log Management From the WebFOCUS Reporting Server Console 10-4
Request Scheduling 10-5
Bulk Loading 10-7
Flat File Targets 10-7
Migrating ETL Requests 10-7
General Workbench Enhancements 10-14
SQL/Answer Set Object 10-15
Source Object 10-15
Target Object 10-16
Validating Records 10-16
Load Options 10-17
Transport Simplification 10-19
Nesting Dependent Requests 10-19
E-mail Request Notification 10-19
Optimized Load 10-21

11. Resource Analyzer and Resource Governor Enhancements 11-1
Configuring Internal Tables From the WebFOCUS Reporting Server Console 11-2
Granting Access to Resource Analyzer/Resource Governor Data Sources 11-3
Running the Resource Governor Configuration Verification Program (CVP) 11-4
Accessing Resource Analyzer and Resource Governor Reports From the WebFOCUS Reporting Server Console 11-6
Increased Support for Long Procedure Names 11-6
Resource Analyzer Reports 11-7
Long-running Requests Report 11-7
Large Volume Request Reports 11-8
Candidates for Column Index Report 11-9

A. New Feature Roadmap for WebFOCUS Version 5 Release 2.3 ..... A-1
WebFOCUS Reporting Language Enhancements A-2
Developer Studio Enhancements A-3
Managed Reporting and Dashboard Enhancements A-4
Ad Hoc Reporting Enhancements A-5
Report Caster Enhancements A-5
WebFOCUS Client Enhancements A-6
WebFOCUS Reporting Server Enhancements A-7
Data Adapter Enhancements A-7
WebFOCUS Maintain Enhancements A-8

CHAPTER 1

WebFOCUS Reporting Language Enhancements

This chapter describes all of the features of the WebFOCUS Reporting language in Version 5 Release 2 and has been updated to reflect Version 5 Release 2.3. Topics that contain Version 5 Release 2.3 features are marked with asterisks.

Topics:

Creating Reports

* Calculating Trends and Predicting Values
  Calculating Cumulative Normal Distribution
  Grouping Numeric Data Into Tiles
* Grouping Sort Field Values in Ranges
* Indicating Total Number of Pages
* Manipulating Summary Values
* Handling Missing Data in HOLD Files
* Using Variable-Length Character Expressions
* Validating Function Arguments
* Financial Reporting Language Enhancements
  Graph Enhancements
* Expanded Limits
  Prompting for Variables in CGI/Servlet
  Including Script Files in Reports
  Specifying Holidays
* Establishing a Non-Standard Alias
* Compiling Expressions Containing DEFINEs
  Exiting on Error
* Converting Case for File Names and Directories

Styling and Displaying Reports

  Adding Borders for Emphasis
  Aligning Decimals in a Multi-Line Heading
  Using Fields to Define Formats
  Punctuating Numbers
  Displaying a Leading Zero in Decimal Numbers
  Selecting a Currency Symbol for Display
* Displaying a Minus Sign After Negative Data
  Displaying Compound Reports in PS Format
  Excel Integration Enhancements
  Selecting a Language for Excel Requests
  Stacking Columns With Multiple FOLD-LINEs
* Specifying Multiple Drill-Down Links
  Navigating Reports From a Table of Contents
  Producing Row Totals for ACROSS Field Values
* Output Formats for Application Processing
  Adjusting Field Displays
  PostScript Type 1 Fonts for PS and PDF Reports
  Controlling the Paper Source for a PS Printer
* SET Parameter Summary
* Functions Summary

Calculating Trends and Predicting Values

In this section:

Calculating Trends and Predicting Values With FORECAST

FORECAST Processing

Using a Simple Moving Average

Using Single Exponential Smoothing

Using Double Exponential Smoothing

Using Triple Exponential Smoothing

Using a Linear Regression Equation

Calculating Trends and Predicting Values Using REGRESS

How to:

Calculate Trends and Predict Values Using FORECAST

You can calculate trends in data, and predict values beyond the range of those stored in the data source, with the FORECAST and REGRESS commands.

- **FORECAST** uses either averages or a linear regression line to distinguish trends in data, using a single independent variable to derive the dependent variable. FORECAST can be used in a report or graph request.
- **REGRESS** uses up to three independent variables to derive a linear equation that best fits a set of numeric data points. This Release 5.2.3 feature is useful when the dependent variable can be approximated by a linear combination of multiple independent variables.

The FORECAST and REGRESS commands cannot be used in the same request.

Calculating Trends and Predicting Values With FORECAST

The calculations you can make to identify trends and forecast values are:

- **Simple moving average (MOVAVE)** calculates a series of arithmetic means using a specified number of values from a field.
- **Exponential moving average** calculates a weighted average between the previously calculated value of the average and the next data point. There are three methods for using an exponential moving average:
  - **Single exponential smoothing (EXPAVE)** calculates an average that allows you to choose weights to apply to newer and older values.
  - **Double exponential smoothing (DOUBLEXP)** accounts for the tendency of data to either increase or decrease over time without repeating.
  - **Triple exponential smoothing (SEASONAL)** accounts for the tendency of data to repeat itself in intervals over time.
- **Linear regression analysis (REGRESS)** derives the coefficients of a straight line that best fits the data points and uses this linear equation to estimate values.

  Note: This calculation is different from multivariate regression, which is described in *Calculating Trends and Predicting Values Using REGRESS* on page 1-18.

When predicting values in addition to calculating trends, the FORECAST command continues the same calculations beyond the data points by using the generated trend values as new data points. For the linear regression technique, the calculated regression equation is used to derive trend and predicted values.

FORECAST performs the calculations based on the data provided, but the user must make decisions about their use and reliability. Therefore, FORECAST predictions are not always reliable, and many factors determine how accurate a prediction will be.

You can conditionally format forecasted values.

For the graphical implementation of these features, see *Calculating Trends and Predicting Values With FORECAST in Developer Studio* in Chapter 2, *Developer Studio Enhancements*.

FORECAST Processing

You invoke FORECAST processing by including FORECAST in a RECAP command. In this command, you specify the parameters needed for generating estimated values, including the field to use in the calculations, the type of calculation to use, and the number of predictions to generate. The RECAP field that contains the result of FORECAST can be a new field (non-recursive) or the same field used in the FORECAST calculations (recursive).

Syntax: How to Calculate Trends and Predict Values Using FORECAST

MOVAVE calculation

```
ON sortfield RECAP result_field[/fmt] = FORECAST(infield, interval,
 npredict, 'MOVAVE', npoint1);
```

EXPAVE calculation

```
ON sortfield RECAP result_field[/fmt] = FORECAST(infield, interval,
 npredict, 'EXPAVE', npoint1);
```

DOUBLEXP calculation

```
ON sortfield RECAP result_field[/fmt] = FORECAST(infield, interval,
 npredict, 'DOUBLEXP', npoint1, npoint2);
```

SEASONAL calculation

```
ON sortfield RECAP result_field[/fmt] = FORECAST(infield, interval,
 npredict, 'SEASONAL', nperiod, npoint1, npoint2, npoint3);
```

REGRESS calculation

```
ON sortfield RECAP result_field[/fmt] = FORECAST(infield, interval,
 npredict, 'REGRESS');
```

where:

sortfield

Is the last ACROSS or BY field in the request. This field must be in numeric or date format. If the request does not contain an ACROSS field, FORECAST works on the last BY field.

result_field

Is the field containing the result of FORECAST. It can be a new field, or the same as *infield*. This must be a numeric field; either a real field, a virtual field, or a calculated field.

Note: The word FORECAST and the opening parenthesis must be on the same line as the syntax *sortfield*=.

fmt

Is the display format for *result_field*. D12.2 is the default value. If *result_field* was previously reformatted using a DEFINE or COMPUTE command, the format specified in the RECAP command is respected.

infield

Is any numeric field. It can be the same field as *result_field* or a different field. It cannot be a date-time field or a numeric field with date display options.

interval

Is the increment to add to each *sortfield* value (after the last data point) to create the next value. This must be a positive integer. To sort in descending order, use the BY HIGHEST phrase. The result of adding this number to the *sortfield* values is converted to the same format as *sortfield*.

For date fields, the minimal component in the format determines how the number is interpreted. For example, if the format is YMD, MDY, or DMY, an interval value of 2 is interpreted as meaning two days; if the format is YM, the 2 is interpreted as meaning two months.

npredict

Is the number of predictions for FORECAST to calculate. It must be an integer greater than or equal to zero. Zero indicates that you do not want predictions, and is only supported with a non-recursive FORECAST. For the SEASONAL method, *npredict* is the number of periods to calculate. The number of points generated is:

```
nperiod * npredict
```

nperiod

For the SEASONAL method, is a positive whole number that specifies the number of data points in a period.

npoint1

Is the number of values to average for the MOVAVE method. For EXPAVE, DOUBLEXP, and SEASONAL, this number is used to calculate the weights for each component in the average. This value must be a positive whole number. The weight, *k*, is calculated by the following formula:

```
k=2/(1+npoint1)
```

npoint2

For DOUBLEXP and SEASONAL, this positive whole number is used to calculate the weights for each term in the trend. The weight, *g*, is calculated by the following formula:

```
g=2/(1+npoint2)
```

npoint3

For SEASONAL, this positive whole number is used to calculate the weights for each term in the seasonal adjustment. The weight, *p*, is calculated by the following formula:

```
p=2/(1+npoint3)
```

Using a Simple Moving Average

Example:

Calculating a Simple Moving Average Column

A simple moving average is a series of arithmetic means calculated with a specified number of values from a field. Each new mean in the series is calculated by dropping the first value used in the prior calculation, and adding the next data value to the calculation.

Simple moving averages are sometimes used to analyze trends in stock prices over time. In this scenario, the average is calculated using a *specified number* of periods of stock prices. A disadvantage to this indicator is that because it drops the oldest values from the calculation as it moves on, it loses its memory over time. Also, mean values are distorted by extreme highs and lows, since this method gives equal weight to each point.

Predicted values beyond the range of the data values are calculated using a moving average that treats the calculated trend values as new data points.

The first complete moving average occurs at the nth data point because the calculation requires *n* values. This is called the lag. The moving average values for the lag rows are calculated as follows: the first value in the moving average column is equal to the first data value, the second value in the moving average column is the average of the first two data values, and so on until the *n*th row, at which point there are enough values to calculate the moving average with the number of values specified.

Example: Calculating a Simple Moving Average Column

The following request defines an integer value named PERIOD to use as the independent variable for the moving average. It predicts three periods of values beyond the range of the retrieved data.

```
DEFINE FILE GGSALES
 SDATE/YYM = DATE;
 SYEAR/Y = SDATE;
 SMONTH/M = SDATE;
 PERIOD/I2 = SMONTH;
END
TABLE FILE GGSALES
 SUM UNITS DOLLARS
 BY CATEGORY BY PERIOD
 WHERE SYEAR EQ 97 AND CATEGORY NE 'Gifts'
 ON PERIOD RECAP MOVAVE/D10.1 = FORECAST(DOLLARS, 1, 3, 'MOVAVE' ,3);
END
```

The output is:

Category	PERIOD	Unit Sales	Dollar Sales	MOVAVE
Coffee	1	61666	801123	801,123.0
	2	54870	682340	741,731.5
	3	61608	765078	749,513.7
	4	57050	691274	712,897.3
	5	59229	720444	725,598.7
	6	58466	742457	718,058.3
	7	60771	747253	736,718.0
	8	54633	655896	715,202.0
	9	57829	730317	711,155.3
	10	57012	724412	703,541.7
	11	51110	620264	691,664.3
	12	58981	762328	702,334.7
	13	0	0	694,975.6
	14	0	0	719,879.4
	15	0	0	705,729.9
Food	1	54394	672727	672,727.0
	2	54894	699073	685,900.0
	3	52713	642802	671,534.0
	4	58026	718514	686,796.3
	5	53289	660740	674,018.7
	6	58742	734705	704,653.0
	7	60127	760586	718,677.0
	8	55622	695235	730,175.3
	9	55787	683140	712,987.0
	10	57340	713768	697,381.0
	11	57459	710138	702,348.7
	12	57290	705315	709,740.3
	13	0	0	708,397.8
	14	0	0	707,817.7
	15	0	0	708,651.9

In the report, the number of values to use in the average is 3 and there are no UNITS or DOLLARS values for the generated PERIOD values.

Each average (MOVAVE value) is computed using DOLLARS values where they exist. The calculation of the moving average begins in the following way:

- The first MOVAVE value (801,123.0) is equal to the first DOLLARS value.
- The second MOVAVE value (741,731.5) is the mean of DOLLARS values one and two: (801,123 + 682,340) /2.
- The third MOVAVE value (749,513.7) is the mean of DOLLARS values one through three: (801,123 + 682,340 + 765,078) /3.
- The fourth MOVAVE value (712,897.3) is the mean of DOLLARS values two through four: (682,340 + 765,078 + 691,274) /3.

For predicted values beyond the supplied values, the calculated MOVAVE values are used as new data points to continue the moving average. The predicted MOVAVE values (starting with 694,975.6 for PERIOD 13) are calculated using the previous MOVAVE values as new data points. For example, the first predicted value (694,975.6) is the average of the data points from periods 11 and 12 (620,264 and 762,328) and the moving average for period 12 (702,334.7). The calculation is: 694,975 = (620,264 + 762,328 + 702,334.7)/3.

Using Single Exponential Smoothing

Example:

Calculating a Single Exponential Smoothing Column

The single exponential smoothing method calculates an average that allows you to choose weights to apply to newer and older values.

The weight given to the newest value is determined by the formula

```
k = 2 / (1 + n)
```

where:

k

Is the newest value.

n

Is an integer greater than one. Increasing *n* increases the weight assigned to the earlier observations (or data instances), as compared to the later ones.

The next calculation of the exponential moving average (EMA) value is derived by the following formula:

```
EMA = (EMA * (1-k)) + (datavalue * k)
```

This means that the newest value from the data source is multiplied by the factor *k* and the current moving average is multiplied by the factor (1-*k*). These quantities are then summed to generate the new EMA.

Note: When the data values are exhausted, the last data value in the sort group is used as the next data value.

Example: Calculating a Single Exponential Smoothing Column

The following request defines an integer value named PERIOD to use as the independent variable for the moving average. It predicts three periods of values beyond the range of retrieved data.

```
DEFINE FILE GGSALES
 SDATE/YYM = DATE;
 SYEAR/Y = SDATE;
 SMONTH/M = SDATE;
 PERIOD/I2 = SMONTH;
END
TABLE FILE GGSALES
 SUM UNITS DOLLARS
 BY CATEGORY BY PERIOD
 WHERE SYEAR EQ 97 AND CATEGORY NE 'Gifts'
 ON PERIOD RECAP EXPAVE/D10.1 = FORECAST(DOLLARS, 1, 3, 'EXPAVE' ,3);
END
```

The output is:

Category	PERIOD	Unit Sales	Dollar Sales	EXPAVE
Coffee	1	61666	801123	801,123.0
	2	54870	682340	741,731.5
	3	61608	765078	753,404.8
	4	57050	691274	722,339.4
	5	59229	720444	721,391.7
	6	58466	742457	731,924.3
	7	60771	747253	739,588.7
	8	54633	655896	697,742.3
	9	57829	730317	714,029.7
	10	57012	724412	719,220.8
	11	51110	620264	669,742.4
	12	58981	762328	716,035.2
	13	0	0	716,035.2
	14	0	0	716,035.2
	15	0	0	716,035.2
Food	1	54394	672727	672,727.0
	2	54894	699073	685,900.0
	3	52713	642802	664,351.0
	4	58026	718514	691,432.5
	5	53289	660740	676,086.3
	6	58742	734705	705,395.6
	7	60127	760586	732,990.8
	8	55622	695235	714,112.9
	9	55787	683140	698,626.5
	10	57340	713768	706,197.2
	11	57459	710138	708,167.6
	12	57290	705315	706,741.3
	13	0	0	706,028.3
	14	0	0	705,671.6
	15	0	0	705,493.3

In the report, three predicted values of EXPAVE are calculated within each value of CATEGORY. For values outside the range of the data, new PERIOD values are generated by adding the interval value (1) to the prior PERIOD value.

Each average (EXPAVE value) is computed using DOLLARS values where they exist. The calculation of the moving average begins in the following way:

- The first EXPAVE value (801,123.0) is the same as the first DOLLARS value.
- The second EXPAVE value (741,731.5) is calculated as follows:

  ```
  n=3 (number used to calculate weights)

  k = 2/(1+n) = 2/4 = 0.5
  ```

  Note that because of rounding and the number of decimal places used, the value derived in this sample calculation varies slightly from the one displayed in the report output:

  ```
  EXPAVE = (EXPAVE*(1-k))+(new-DOLLARS*k) = (801123*0.5) + (682340*0.50)
         =  400561.5 + 341170 = 741731.5
  ```

- The third EXPAVE value (753,404.8) is calculated as follows:

  ```
  EXPAVE = (EXPAVE*(1-k))+(new-DOLLARS*k) = (741731.5*0.5)+(765078*0.50)
         =  370865.75 + 382539 = 753404.75
  ```

For predicted values beyond those supplied, the last EXPAVE value is used as the new data point in the exponential smoothing calculation. The predicted EXPAVE values (starting with 706,741.6) are calculated using the previous average and the new data point. Because the previous average is also used as the new data point, the predicted values are always equal to the last trend value. For example, the previous average for period 13 is 706,741.6, and this is also used as the next data point. Therefore, the average is calculated as follows: (706741.6 * 0.5) + (706741.6 * 0.5) = 706741.6

```
EXPAVE = (EXPAVE * (1-k)) + (new-DOLLARS * k) = (706741.6*0.5) +
         (706741.6*0.50) = 353370.8 + 353370.8 = 706741.6
```

Using Double Exponential Smoothing

Example:

Calculating a Double Exponential Smoothing Column

Double exponential smoothing produces an exponential moving average that takes into account the tendency of data to either increase or decrease over time without repeating. This is accomplished by using two equations with two constants:

- The first equation accounts for the current time period and is a weighted average of the current data value and the prior average, with an added component (*b*) that represents the trend for the previous period. The weight constant is *k*:

  ```
  DOUBLEXP(t) = k * datavalue(t) + (1-k) * ((DOUBLEXP(t-1) + b(t-1))
  ```

- The second equation is the calculated trend value, and is a weighted average of the difference between the current and previous average and the trend for the previous time period. *b*(*t*) represents the average trend. The weight constant is *g*:

  ```
  b(t) = g * (DOUBLEXP(t)-DOUBLEXP(t-1)) + (1 - g) * (b(t-1))
  ```

These two equations are solved to derive the smoothed average. The first smoothed average is set to the first data value. The first trend component is set to zero. For choosing the two constants, the best results are usually obtained by minimizing the mean-squared error (MSE) between the data values and the calculated averages. You may need to use nonlinear optimization techniques to find the optimal constants.

The equation used for forecasting beyond the data points with double exponential smoothing is:

```
forecast(t+m) = DOUBLEXP(t) + m * b(t)
```

where:

m

Is the number of time periods ahead for the forecast.

Example: Calculating a Double Exponential Smoothing Column

The following sums the ACTUAL_YTD field of the CENTSTMT data source by period, and calculates a single exponential and double exponential moving average.

```
TABLE FILE CENTSTMT
SUM ACTUAL_YTD
 BY PERIOD
 ON PERIOD RECAP EXP/D15.1 = FORECAST(ACTUAL_YTD, 1, 0, 'EXPAVE' ,3);
 ON PERIOD RECAP DOUBLEXP/D15.1 = FORECAST(ACTUAL_YTD, 1, 0,
  'DOUBLEXP', 3, 3);
 WHERE GL_ACCOUNT LIKE '3%%%'
END
```

The output is:

	YTD		
PERIOD	Actual	EXP	DOUBLEXP
2002/01	12,957,681.	12,957,681.0	12,957,681.0
2002/02	25,441,971.	19,199,826.0	22,439,246.3
2002/03	39,164,321.	29,182,073.5	34,791,885.1
2002/04	52,733,326.	40,957,699.8	48,845,816.0
2002/05	66,765,920.	53,861,809.9	63,860,955.9
2002/06	80,952,492.	67,407,150.9	79,188,052.9

Using Triple Exponential Smoothing

Example:

Calculating a Triple Exponential Smoothing Column

Triple exponential smoothing produces an exponential moving average that takes into account the tendency of data to repeat itself in intervals over time. For example, sales data that is growing and in which 25% of sales always occur during December contains both trend and seasonality. Triple exponential smoothing takes both the trend and seasonality into account by using three equations with three constants.

Note: For triple exponential smoothing, you must know the number of data points in each time period (designated as L in the following equations). To account for the seasonality, a seasonal index is calculated. The data is divided by the prior season's index and then used in calculating the smoothed average.

- The first equation accounts for the current time period, and is a weighted average of the current data value divided by the seasonal factor and the prior average adjusted for the trend for the previous period. The weight constant is *k*:

  ```
  SEASONAL(t) = k * (datavalue(t)/I(t-L)) + (1-k) * (SEASONAL(t-1) +
  b(t-1))
  ```

- The second equation is the calculated trend value, and is a weighted average of the difference between the current and previous average and the trend for the previous time period. *b*(*t*) represents the average trend. The weight constant is *g*:

  ```
  b(t) = g * (SEASONAL(t)-SEASONAL(t-1)) + (1-g) * (b(t-1))
  ```

- The third equation is the calculated seasonal index, and is a weighted average of the current data value divided by the current average and the seasonal index for the previous season. *I*(*t*) represents the average seasonal coefficient. The weight constant is *p*:

  ```
  I(t) = p * (datavalue(t)/SEASONAL(t)) + (1 - p) * I(t-L)
  ```

These equations are solved to derive the triple smoothed average. The first smoothed average is set to the first data value. Initial values for the seasonality factors are calculated based on the maximum number of full periods of data in the data source, while the initial trend is calculated based on two periods of data. These values are calculated with the following steps:

1. The initial trend factor is calculated by the following formula:

   ```
   b(0) = (1/L) ((y(L+1)-y(1))/L + (y(L+2)-y(2))/L + ... + (y(2L) -
   y(L))/L )
   ```

2. The calculation of the initial seasonality factor is based on the average of the data values within each period, *A*(*j*) (1<=*j*<=*N*):

   ```
   A(j) = ( y((j-1)L+1) + y((j-1)L+2) + ... + y(jL) ) / L
   ```

3. Then, the initial periodicity factor is given by the following formula, where *N* is the number of full periods available in the data, *L* is the number of points per period and *n* is a point within the period (1<= *n* <= *L*):

   ```
   I(n) = ( y(n)/A(1) + y(L+n)/A(2) + ... + y((N-1)L+n)/A(N) ) / N
   ```

The three constants must be chosen carefully. The best results are usually obtained by choosing the constants to minimize the mean-squared error (MSE) between the data values and the calculated averages. You may need to use nonlinear optimization techniques to find the optimal constants.

The equation used to forecast beyond the last data point with triple exponential smoothing is

```
forecast(t+m) = (SEASONAL(t) + m * b(t)) / I(t-L+MOD(m/L))
```

where:

`m`

Is the number of periods ahead for the forecast.

Example: Calculating a Triple Exponential Smoothing Column

In the following request, the data has seasonality but no trend. Therefore, *npoint2* is set high (1000) to make the trend factor negligible in the calculation:

```
SET HISTOGRAM = OFF
 TABLE FILE VIDEOTRK
 SUM TRANSTOT
 BY  TRANSDATE
 ON TRANSDATE RECAP SEASONAL/D10.1 = FORECAST(TRANSTOT, 1, 3, 'SEASONAL',
  3, 3, 1000, 1);
 WHERE TRANSDATE NE '19910617'
END
```

In the output, *npredict* is 3. Therefore, three periods (nine points, *nperiod* * *npredict*) are generated.

TRANSDATE	TRANSTOT	SEASONAL
91/06/18	21.25	21.3
91/06/19	38.17	31.0
91/06/20	14.23	34.6
91/06/21	44.72	53.2
91/06/24	126.28	75.3
91/06/25	47.74	82.7
91/06/26	40.97	73.7
91/06/27	60.24	62.9
91/06/28	31.00	66.3
91/06/29		45.7
91/06/30		94.1
91/07/01		53.4
91/07/02		72.3
91/07/03		140.0
91/07/04		75.8
91/07/05		98.9
91/07/06		185.8
91/07/07		98.2

Using a Linear Regression Equation

Example:
Calculating a Linear Regression Field

The Linear Regression Equation estimates values by assuming that the dependent variable (the new calculated values) and the independent variable (the sort field values) are related by a function that represents a straight line.

The equation is

```
y = mx + b
```

where:

`y`
Is the dependent variable.

`m`
Is the slope of the line.

`x`
Is the independent variable.

`b`
Is the y-intercept.

This calculation uses a technique called Ordinary Least Squares to determine values for *m* and *b* that minimize the sum of the squared differences between the data and the resulting line.

The following formulas show how *m* and *b* are calculated.

$$m = \frac{(\sum xy - (\sum x \bullet \sum y)/n)}{\left(\sum x^2 - (\sum x)^2/n\right)}$$

$$b = (\sum y)/n - (m \bullet (\sum x)/n)$$

where:

`x`
Are the sort field values (independent variables).

`y`
Are the data values (dependent variables).

`n`
Is the number of data points.

Trend values as well as predicted values are calculated using the regression line equation.

Example: Calculating a Linear Regression Field

```
TABLE FILE CAR
 PRINT MPG
 BY DEALER_COST
 WHERE MPG NE 0.0
 ON DEALER_COST RECAP FORMPG = FORECAST(MPG, 1000, 3, 'REGRESS');
END
```

The output is:

DEALER_COST	MPG	FORMPG
2,886	27	25.51
4,292	25	23.65
4,631	21	23.20
4,915	21	22.82
5,063	23	22.63
5,660	21	21.83
	21	21.83
5,800	24	21.65
6,000	24	21.38
7,427	16	19.49
8,300	18	18.33
8,400	18	18.20
10,000	18	16.08
11,000	18	14.75
11,194	9	14.50
14,940	11	9.53
15,940	0	8.21
16,940	0	6.88
17,940	0	5.55

Note:

- Three predicted values of FORMPG are calculated. For values outside the range of the data, new DEALER_COST values are generated by adding the interval value (1,000) to the prior DEALER_COST value.
- There are no MPG values for the generated DEALER_COST values.
- Each FORMPG value is computed using a regression line, calculated using all of the actual data values for MPG.

  DEALER_COST is the independent variable (*x*) and MPG is the dependent variable (*y*). The equation is used to calculate FORMPG trend and predicted values.

In this case, the equation is approximately as follows:

```
FORMPG = (-0.001323 * DEALER_COST) + 29.32
```

The predicted values are (the values are not exactly as calculated by FORECAST because of rounding, but they show the calculation process):

DEALER_COST	Calculation	FORMPG
15,940	(-0.001323 * 15,940) + 29.32	8.23
16,940	(-0.001323 * 16,940) + 29.32	6.91
17,940	(-0.001323 * 17,940) + 29.32	5.59

Calculating Trends and Predicting Values Using REGRESS

How to:

Calculate Trends and Predict Values Using REGRESS

Example:

Creating a Multivariate Linear Regression Column Using REGRESS

In Release 5.2.3, you can calculate trends and predict values with multivariate regression. This method derives a linear equation that best fits a set of numeric data points, and uses this equation to create a new column in the report output. The equation can be based on one to three independent variables.

This method estimates values by assuming that the dependent variable (*y*, the new calculated values) and the independent variables (*x1*, *x2*, *x3*) are related by the following linear equation:

```
y = a1*x1 [+ a2*x2 [+ a3*x3]] + b
```

When there is one independent variable, the equation represents a straight line. This produces the same values as FORECAST using the linear regression (REGRESS) method. When there are two independent variables, the equation represents a plane, and with three independent variables, it represents a hyperplane. You should use this technique when you have reason to believe that the dependent variable can be approximated by a linear combination of the independent variables.

REGRESS uses a technique called Ordinary Least Squares to calculate values for the coefficients (*a1*, *a2*, *a3*, and *b*) that minimize the sum of the squared differences between the data and the resulting line, plane, or hyperplane.

Syntax: How to Calculate Trends and Predict Values Using REGRESS

```
ON {sortfield|TABLE} RECAP y[/fmt] = REGRESS(n, x1, [x2, [x3,]] z);
```

where:

sortfield

Is a field in the data source. It cannot be the same field as any of the parameters to REGRESS. A new linear regression equation is derived each time the sort field value changes.

y

Is the new numeric column calculated by applying the regression equation. You cannot DEFINE or COMPUTE a field with this name.

fmt

Is the display format for *y*. If it is omitted, D12.2 is the default value. Format I is not supported for *y*.

n

Is a whole number from 1 to 3 indicating the number of independent variables.

x1, x2, x3

Are the field names to be used as the independent variables. All of these variables must be numeric and be independent of each other.

z

Is an existing numeric field that is assumed to be approximately linearly dependent on the independent variables and is used to derive the regression equation.

Example: Creating a Multivariate Linear Regression Column Using REGRESS

The following request uses the GGSALES data source to calculate an estimated DOLLARS column. The BUDUNITS, UNITS, and BUDDOLLARS fields are the independent variables. The DOLLARS field provides the actual values to be estimated:

```
DEFINE FILE GGSALES
 SDATE/YYM = DATE;
 SYEAR/Y = SDATE;
END
TABLE FILE GGSALES
 PRINT BUDUNITS UNITS BUDDOLLARS DOLLARS
 ON TABLE RECAP ESTDOLLARS/F8 = REGRESS(3, BUDUNITS, UNITS,
  BUDDOLLARS, DOLLARS);
 WHERE SYEAR EQ 97
 WHERE CATEGORY EQ 'Coffee'
 WHERE REGION EQ 'West'
 WHERE UNITS GT 1600 AND UNITS LT 1700
END
```

The output is:

```
Budget Units  Unit Sales  Budget Dollars  Dollar Sales  ESTDOLLARS
------------  ----------  --------------  ------------  ----------
        1796        1696           17960         25440       23381
        1463        1663           21945         19966       18919
        1759        1615           17590         17765       20850
        1403        1601           21045         17611       16530
        1825        1695           23725         25425       24316
        1613        1685           22582         18535       21365
        1568        1682           23520         25230       20841
        1733        1647           17330         24705       21313
        1870        1620           24310         24300       23033
        1665        1678           21645         23492       21722
        1725        1669           22425         21697       22307
        1457        1671           21855         20052       19038
        1662        1674           24930         18414       21931
        1556        1643           18672         18073       19201
        1653        1694           21489         16940       21963
        1473        1616           14730         16160       17095
```

Calculating Cumulative Normal Distribution

How to:

Calculate Cumulative Standard Normal Distribution

Calculate Inverse Cumulative Standard Normal Distribution

Reference:

Characteristics of a Standard Normal Distribution Curve

Example:

Using the NORMSDST and NORMSINV Functions

You can perform calculations on a standard normal distribution curve with the NORMSDST and NORMSINV functions:

- The NORMSDST function calculates the percentage of data values that are less than or equal to a normalized value. A normalized value is a point on the X-axis of a standard normal distribution curve. This is useful for determining percentiles in normally distributed data.
- The NORMSINV function finds the normalized value that forms the upper boundary of a percentile in a standard normal distribution curve. NORMSINV is the inverse of NORMSDST.

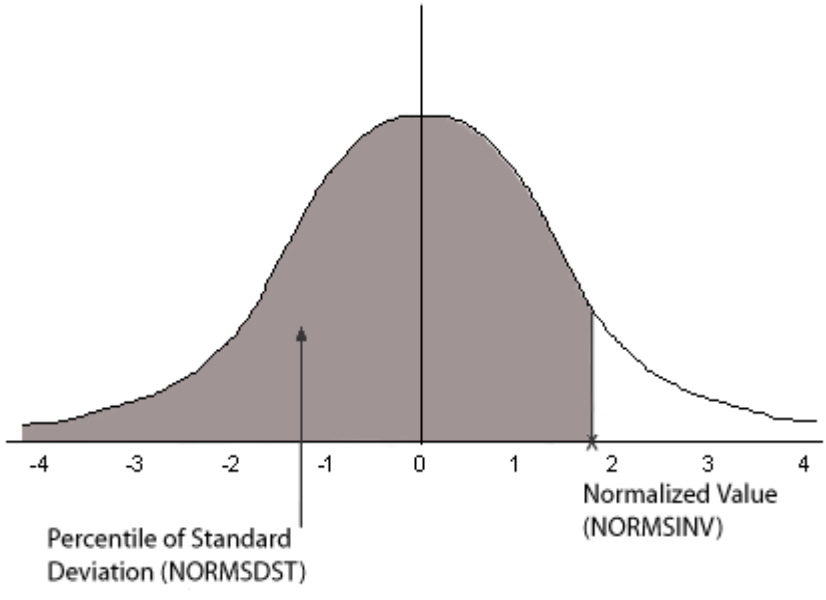
The results of NORMSDST and NORMSINV are returned as double precision and are accurate to six significant digits.

Information Builders-supplied functions are fully documented in the *WebFOCUS Using Functions* manual.

Reference: Characteristics of a Standard Normal Distribution Curve

A standard normal distribution curve is a normal distribution that has a mean of 0 and a standard deviation of 1. The total area under this curve is 1. A point on the X-axis of the standard normal distribution is called a normalized value. Assuming that your data is normally distributed, you can convert a data point to a normalized value in order to find the percentage of scores that are less than or equal to the raw score.

The results of the NORMSDST and NORMSINV functions are illustrated by the following diagram:



Many common measurements tend to be normally distributed. A plot of normally distributed data values approximates a bell-shaped curve. The two measures needed to describe any normal distribution are the mean and the standard deviation:

- *Mean* is the point at the center of the curve.
- *Standard deviation* describes the spread of the curve. It is the distance from the mean to the point of inflection (the point where the curve changes direction).

Syntax: How to Calculate Cumulative Standard Normal Distribution

```
NORMSDST(value, 'D8');
```

where:

value

Is a normalized value.

'D8'

Is the required format for the result. The value returned by the function is double precision. You can assign it to a field with any valid numeric format.

Syntax: How to Calculate Inverse Cumulative Standard Normal Distribution

```
NORMSINV(value, 'D8');
```

where:

value

Is a number between 0 and 1 which represents the percentile in a standard normal distribution.

'D8'

Is the required format for the result. The value returned by the function is double precision. You can assign it to a field with any valid numeric format.

Example: Using the NORMSDST and NORMSINV Functions

The NORMSDST function finds the percentile for the Z field. The NORMSINV function then returns this percentile to a normalized value.

```
DEFINE FILE GGPRODS
 -* CONVERT SIZE FIELD TO DOUBLE PRECISION
 X/D12.5 = SIZE;
END
TABLE FILE GGPRODS
 SUM X NOPRINT CNT.X NOPRINT
 -* CALCULATE MEAN AND STANDARD DEVIATION
 COMPUTE NUM/D12.5 = CNT.X; NOPRINT
 COMPUTE MEAN/D12.5 = AVE.X; NOPRINT
 COMPUTE VARIANCE/D12.5 = ((NUM*ASQ.X) - (X*X/NUM))/(NUM-1); NOPRINT
 COMPUTE STDEV/D12.5 = SQRT(VARIANCE); NOPRINT
 PRINT SIZE X NOPRINT
 -* COMPUTE NORMALIZED VALUES AND USE AS INPUT TO NORMSDST FUNCTION
 -* THEN USE RETURNED VALUES AS INPUT TO NORMSINV FUNCTION
 -* AND CONVERT BACK TO DATA VALUES
 COMPUTE Z/D12.5 = (X - MEAN)/STDEV;
 COMPUTE NORMSD/D12.5 = NORMSDST(Z, 'D8');
 COMPUTE NORMSI/D12.5 = NORMSINV(NORMSD, 'D8');
 COMPUTE DSIZE/D12 = NORMSI * STDEV + MEAN;
 BY PRODUCT_ID NOPRINT
END
```

The output is:

Size	Z	NORMSD	NORMSI	DSIZE
16	-.07298	.47091	-.07298	16
12	-.80273	.21106	-.80273	12
12	-.80273	.21106	-.80273	12
20	.65678	.74434	.65678	20
24	1.38654	.91721	1.38654	24
20	.65678	.74434	.65678	20
24	1.38654	.91721	1.38654	24
16	-.07298	.47091	-.07298	16
12	-.80273	.21106	-.80273	12
8	-1.53249	.06270	-1.53249	8

Grouping Numeric Data Into Tiles

How to:

Group Numeric Data Into Tiles

Example:

Grouping Data Into Tiles

You can group numeric data into any number of tiles (percentiles, quartiles, deciles, and so on) in tabular reports. For example, you can group student test scores into deciles to determine which students are in the top ten percent of the class.

Grouping is based on the values in the selected vertical (BY) sort field, and is apportioned as equally as possible into the number of tile groups you specify.

Syntax: How to Group Numeric Data Into Tiles

```
BY [{HIGHEST|LOWEST} [k]] tilefield [AS 'head1']
 IN-GROUPS-OF n TILES [TOP m] [AS 'head2']
```

where:

`HIGHEST`

Sorts the data in descending order so that the highest data values are placed in tile 1.

`LOWEST`

Sorts the data in ascending order so that the lowest data values are placed in tile 1. LOWEST is the default value.

k

Is a positive integer representing the number of tile groups to display in the report. For example, BY HIGHEST 2 displays the two non-empty tiles with the highest data values.

tilefield

Is the field whose values are used to assign the tile numbers.

head1

Is a heading for the column that displays the values of the tile sort field.

n

Is a positive integer not greater than 32,767 specifying the number of tiles to be used in grouping the data. For example, 100 tiles produces percentiles, 10 tiles produces deciles.

m

Is a positive integer indicating the highest tile value to display in the report. For example, TOP 3 does not display any data row that is assigned a tile number greater than 3.

head2

Is a new heading for the column that displays the tile numbers.

Note:

- The syntax accepts numbers that are not integers for *k*, *n*, and *m*. On OS/390 and z/OS and VM, values with decimals are rounded to integers; on UNIX and Windows NT, they are truncated. If the numbers supplied are negative or zero, a message is generated.
- Both *k* and *m* limit the number of rows displayed within each sort break in the report. If you specify both, the more restrictive value will control the display. If *k* and *m* are both greater than *n* (the number of tiles), *n* will be used.

Example: Grouping Data Into Tiles

The following request illustrates group data into five tiles.

```
TABLE FILE MOVIES
 PRINT TITLE
 BY CATEGORY
 BY LISTPR IN-GROUPS-OF 5 TILES
 WHERE CATEGORY EQ 'ACTION' OR 'CHILDREN'
END
```

The output is:

```
CATEGORY LISTPR TILE TITLE
ACTION     14.95    1 TOP GUN
           19.95    2 JAWS
                      RAMBO III
           19.98    4 ROBOCOP
           19.99    5 TOTAL RECALL
CHILDREN   14.95    1 SESAME STREET-BEDTIME STORIES AND SONGS
           14.98    1 ROMPER ROOM-ASK MISS MOLLY
           19.95    2 SMURFS, THE
                      SCOOBY-DOO-A DOG IN THE RUFF
           26.99    3 BAMBI
           29.95    4 ALICE IN WONDERLAND
                      SLEEPING BEAUTY
           44.95    5 SHAGGY DOG, THE
```

Note that the tiles are assigned within the higher-level sort field CATEGORY. The ACTION category does not have any data assigned to tile 3. The CHILDREN category has all five tiles.

For additional examples, see *Grouping Numeric Data Into Tiles* in the *Sorting Tabular Reports* chapter in the *Creating Reports With WebFOCUS Language* manual.

For two graphical implementations of this feature, see the following topics in the C*reating Reports With Graphical Tools* manual:

- For Report Painter, *How to Group Data Into Tiles* in the *Creating Reports With the Report Painter* chapter.
- For an OLAP report, *Grouping Numeric Data Into Tiles* in the *Analyzing Data in an OLAP Report* chapter.

Grouping Sort Field Values in Ranges

How to:

Define Equal Ranges

Example:

Defining Equal Ranges

In Release 5.2.3, when you sort a report using a numeric sort field, you can group the sort field values together and define groups of equal range using the IN-RANGES-OF phrase.

Each report request can contain a total of five IN-GROUPS-OF phrases plus IN-RANGES-OF phrases. The IN-RANGES-OF phrase can be used only once per BY field. The first sort field range starts from the lowest value of a multiple of the IN-GROUPS-OF value.

For a graphical implementation of this feature, see *How to Group Numeric Data Into Ranges* in the *Creating Reports With Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.

Syntax: How to Define Equal Ranges

```
{BY|ACROSS} sortfield IN-RANGES-OF value [TOP limit]
```

where:

sortfield

Is the name of the sort field. The sort field must be numeric. Its format must be I (integer), F (floating point), D (double precision), or P (packed).

value

Is an integer greater than zero, indicating the range by which sort field values are grouped.

limit

Is an optional number that defines the highest range label to be included in the report. The range is extended to include all data values higher than this value.

Example: Defining Equal Ranges

This following request creates five groups of equal range:

```
TABLE FILE EMPLOYEE
 PRINT LAST_NAME
 BY CURR_SAL IN-RANGES-OF 5000
END
```

The output is:

```
CURR SAL              LAST NAME
$5,000.00 - $9,999.99 SMITH
                      GREENSPAN
$10,000.00 - $14,999.99 STEVENS
                      SMITH
$15,000.00 - $19,999.99 JONES
                      MCCOY
                      MCKNIGHT
$20,000.00 - $24,999.99 ROMANS
                      BLACKWOOD
$25,000.00 - $29,999.99 BANNING
                      IRVING
                      CROSS
```

Indicating Total Number of Pages

How to:

Display Page X of Y in a Report

Example:

Displaying Page X of Y

You can use the TABLASTPAGE system variable to insert the total number of pages in an HTML, PDF, PS, or Excel 2000 report. For example, you can add a footing that reads "Page 1 of 5", where the value of TABLASTPAGE is 5.

Syntax: How to Display Page X of Y in a Report

```
heading_or_footing_phrase
"Page <TABPAGENO of <TABLASTPAGE"
  .
  .
  .
ON TABLE HOLD AS name FORMAT fmt
```

where:

heading_or_footing_phrase

Can be one of the following:

```
HEADING [CENTER]
FOOTING [CENTER]
ON sortfield SUBHEAD
ON sortfield SUBFOOT
```

where *sortfield* is the sort field that triggers a new subheading or subfooting.

name

Is the name of the generated PDF, Excel 2000, PostScript, or HTML file.

fmt

Can be one of the following:

```
PDF
EXL2K
PS
POSTSCRIPT
HTML
```

Example: Displaying Page X of Y

```
TABLE FILE GGSALES
 "Page <TABPAGENO of <TABLASTPAGE"
 " "
 SUM UNITS AS 'Unit,Sales'
 BY CATEGORY BY REGION BY PRODUCT
 ON CATEGORY PAGE-BREAK
 ON TABLE HOLD FORMAT HTML
 ON TABLE SET STYLE *
 TYPE = REPORT, GRID = OFF ,$
 TYPE = REPORT, FONT = HELVETICA ,$
 TYPE = HEADING, STYLE = ITALIC ,$
 ENDSTYLE
END
```

The output is:

Page 1 of 3

Category	Region	Product	Unit Sales
Coffee	Midwest	Espresso	101154
		Latte	231623
	Northeast	Capuccino	44785
		Espresso	68127
		Latte	222866
	Southeast	Capuccino	73264
		Espresso	68030
		Latte	209654
	West	Capuccino	71168
		Espresso	71675
		Latte	213920

Page 2 of 3

Category	Region	Product	Unit Sales
Food	Midwest	Biscotti	86105
		Croissant	139182
		Scone	116127
	Northeast	Biscotti	145242
		Croissant	137394
		Scone	70732
	Southeast	Biscotti	119594
		Croissant	156456
		Scone	73779
	West	Biscotti	70436
		Croissant	197022
		Scone	72776

Page 3 of 3

Category	Region	Product	Unit Sales
Gifts	Midwest	Coffee Grinder	50393
		Coffee Pot	47156
		Mug	86718
		Thermos	46587
	Northeast	Coffee Grinder	40977
		Coffee Pot	46185
		Mug	91497
		Thermos	48870
	Southeast	Coffee Grinder	47083
		Coffee Pot	49922
		Mug	88474
		Thermos	48976
	West	Coffee Grinder	48081
		Coffee Pot	47432
		Mug	93881
		Thermos	45648

Manipulating Summary Values

In this section:

Controlling Summary Line Processing

Using Prefix Operators With Calculated Values

Using Multiple SUB-TOTAL or SUMMARIZE Commands

How to:

Use Prefix Operators With Summary Values

Example:

Using Prefix Operators With SUBTOTAL at the Sort Break

Displaying an Alphanumeric Field on a Summary Line

You can use the SUBTOTAL, SUB-TOTAL, RECOMPUTE, and SUMMARIZE commands at the ON TABLE level to specify the type of summary operation to use to produce the grand total line on the report.

Prefix operations on summary lines are performed on the retrieved, selected, and summed values that will become the detail lines in the report. They are not performed on each incoming record (as field-based prefix operations are).

Each type of summary has its own purpose and handles the prefix operators appropriately for the type of summary information to be displayed. For example, using AVE. at a sort field break produces the average within the sort group.

In Release 5.2.3, you can also display alphanumeric fields on summary lines. To display an alphanumeric field on a summary line, you must either explicitly list its field name on the summary command or use the asterisk (*) wildcard to display all fields.

Different operations from two ON phrases for the same sort break appear on the same summary line and allow a mixture of operations on summary lines. The grand total line populates all fields populated by any summary command, even fields that are not specified in the grand total command.

If the same field is referenced in more than one ON phrase for the same sort break, the last function specified is applied.

The following prefix operators are supported for numeric fields:

ASQ	LST
AVE	MAX
CNT	MIN
FST	SUM

The following prefix operators are supported for alphanumeric fields:

FST	MIN
LST	SUM (means LST. if SUMPREFIX = LST; means FST. if SUMPREFIX = FST)
MAX	

Several examples of the use of prefix operators with summary values are shown here. For additional examples, see *Manipulating Summary Values With Prefix Operators* in the *Including Totals and Subtotals* chapter in the *Creating Reports With WebFOCUS Language* manual.

Syntax: How to Use Prefix Operators With Summary Values

```
{BY|ON} sortfield [AS 'text1'] summaryoption [MULTILINES]
 operator_and_field_list [AS 'text2'] [WHEN expression;]
```

where:

sortfield

Is a sort field. When the value of the sort field changes, it triggers the summary operation.

text1

Is the column heading to use for the sort field on the report output.

summaryoption

Is one of the following: SUBTOTAL, SUB-TOTAL, RECOMPUTE, or SUMMARIZE.

MULTILINES

Suppresses the printing of a summary line for every sort break that has only one detail line. Note that MULTILINES suppresses the summary line even if a prefix operator is used to specify a different operation for the summary line. MULTI-LINES is a synonym for MULTILINES.

operator_and_field_list

Can have one of the following forms:

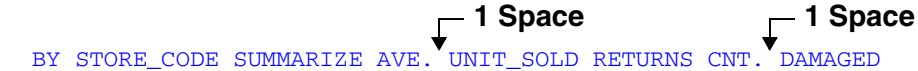
*[prefixop.] ** includes all display fields on the summary line. If a prefix operator is specified, it is applied to all fields. If the prefix operator is not supported with alphanumeric fields, alphanumeric fields are not included on the summary line.

prefixop. is applied to every numeric column in the report output. Every numeric column is populated with values on the summary row.

prefixop1. field1 [field2 ... fieldn] [prefixop2. fieldm ...] is applied as follows: the first prefix operator is applied to *field1* through *fieldn*; the second prefix operator is applied to *fieldm*. *field1* to *fieldm* are the field names to be displayed for the summary line for the specified sort break. Each field name must be preceded by a space.

Only the fields specified are populated with values on the summary row.

Each prefix operator must be separated from the field name by a space. For example:



text2

Is the text that prints on the left of the summary row.

expression

Is an expression that determines whether the summary operation is performed at each break.

Example: Using Prefix Operators With SUBTOTAL at the Sort Break

The following request uses prefix operators to calculate the:

- Average list price by rating within category.
- Sum of copies by category.

Notice that the subtotal row for each rating contains a value only in the LISTPR column and the subtotal row for each category contains a value only in the COPIES column. The default grand total line contains values only for the columns that were subtotaled. Note the blank space between each prefix operator and the field name that follows it:

```
TABLE FILE MOVIES
 PRINT COPIES LISTPR WHOLESALEPR TITLE/A23
 BY RATING BY CATEGORY
 WHERE CATEGORY EQ 'CHILDREN' OR 'CLASSIC'
 WHERE RATING EQ 'G' OR 'NR'
 ON RATING SUBTOTAL AVE. LISTPR AS '*Ave: '
 ON CATEGORY SUBTOTAL SUM. COPIES AS '*Sum: '
END
```

The output is:

RATING	CATEGORY	COPIES	LISTPR	WHOLESALEPR	TITLE
G	CHILDREN	2	44.95	29.99	SHAGGY DOG, THE
		2	29.95	12.50	ALICE IN WONDERLAND
		3	26.99	12.00	BAMBI
*Sum: CHILDREN		7			
	CLASSIC	3	89.95	40.99	GONE WITH THE WIND
*Sum: CLASSIC		3			
*Ave: G			47.96		
NR	CHILDREN	1	19.95	10.00	SMURFS, THE
		1	19.95	9.75	SCOOBY-DOO-A DOG IN THE
		1	14.95	7.65	SESAME STREET-BEDTIME S
		1	14.98	7.99	ROMPER ROOM-ASK MISS MO
		1	29.95	15.99	SLEEPING BEAUTY
*Sum: CHILDREN		5			
	CLASSIC	1	24.98	14.99	EAST OF EDEN
		3	39.99	20.00	CITIZEN KANE
		1	29.95	15.99	CYRANO DE BERGERAC
		1	19.99	10.95	MARTY
		2	19.99	10.95	MALTESE FALCON, THE
		2	19.95	9.99	ON THE WATERFRONT
		2	89.99	40.99	MUTINY ON THE BOUNTY
		2	19.99	10.95	PHILADELPHIA STORY, THE
		2	19.98	10.99	CAT ON A HOT TIN ROOF
		2	29.95	15.00	CASABLANCA
*Sum: CLASSIC		18			
*Ave: NR			27.64		
TOTAL		33	31.91		

Example: Displaying an Alphanumeric Field on a Summary Line

The following request displays the sum of the list price field and the minimum value of the director field by rating:

```
TABLE FILE MOVIES
 PRINT COPIES LISTPR WHOLESALEPR DIRECTOR
 BY RATING
 BY CATEGORY
 WHERE CATEGORY EQ 'CHILDREN' OR 'CLASSIC'
 WHERE RATING EQ 'G' OR 'NR'
 WHERE DIRECTOR NE ' '
 ON RATING SUBTOTAL SUM. LISTPR MIN. DIRECTOR AS '*A/N:'
END
```

The output is:

RATING	CATEGORY	COPIES	LISTPR	WHOLESALEPR	DIRECTOR
G	CHILDREN	2	44.95	29.99	BARTON C.
		2	29.95	12.50	GEROMINI
		3	26.99	12.00	DISNEY W.
	CLASSIC	3	89.95	40.99	FLEMING V.
*A/N: G			191.84		BARTON C.
NR	CHILDREN	1	29.95	15.99	DISNEY W.
	CLASSIC	1	24.98	14.99	KAZAN E.
		3	39.99	20.00	WELLES O.
		1	29.95	15.99	GORDON M.
		1	19.99	10.95	MANN D.
		2	19.99	10.95	HUSTON J.
		2	19.95	9.99	KAZAN E.
		2	89.99	40.99	MILESTONE L.
		2	19.99	10.95	CUKOR G.
		2	19.98	10.99	BROOKS R.
		2	29.95	15.00	CURTIZ M.
*A/N: NR			344.71		BROOKS R.
TOTAL			536.55		BARTON C.

Controlling Summary Line Processing

How to:

Control Summary Line Processing

Reports that use prefix operators on summary lines are processed differently than those without prefix operators. In some cases, a different style of report output results from each type of request.

If a summary command specifies one field name and another summary command specifies a second field name:

- In a report without summary prefix operators, both columns are populated on both summary lines.
- In a report with summary prefix operators, only the specified column is populated on each summary line.

If a prefix operator is used in any summary command, prefix operator processing is required for the request. In most requests, it is clear which type of processing to use even if prefix operators are specified in some summary commands but not in others.

However, if the first use of a prefix operator occurs after a field name has been specified in a summary command *without* an accompanying prefix operator, neither type of processing can be implemented. Processing stops, and the following message is generated:

```
(FOC36376) CANNOT COMBINE SUBTOTAL/RECOMPUTE STYLES WHEN SUMMARYLINES=OLD
```

For example:

```
ON RATING SUBTOTAL COPIES AVE. LISTPR
```

or

```
ON RATING SUBTOTAL LISTPR
ON CATEGORY SUBTOTAL AVE. COPIES
```

To allow this combination, you can issue the SET SUMMARYLINES = NEW command to specify that prefix operator processing should be used. The SUM. operator is then applied to any field that does have a prefix operator.

Syntax: How to Control Summary Line Processing

```
SET SUMMARYLINES = {OLD|NEW}
```

where:

OLD

Does not allow mixing of summary fields with and without prefix operators when the first field name used in a summary command does not have an associated prefix operator. OLD is the default value.

You can specify the SUM. operator for fields for which you want a standard subtotal. This produces the same value that would have been generated without prefix operators.

NEW

Allows mixing of summary fields with and without prefix operators even when the first field name specified in a summary command does not have a prefix operator. All summary fields without prefix operators are processed as though they were specified with the SUM. operator.

Using Prefix Operators With Calculated Values

Example:

Using Prefix Operators With RECOMPUTE

If a request includes the RECOMPUTE or SUMMARIZE command, the expression specified in the associated COMPUTE command is applied using values from the summary line. The columns used to recompute the expression can have prefix operators. The recomputed column, regardless of the prefix operator specified for it, applies these input values to the expression specified in the COMPUTE command. Therefore, any supported prefix operator can be specified for the recomputed report column without affecting the calculated result.

With prefix operator processing, all fields used in the COMPUTE command must be calculated by the RECOMPUTE or SUMMARIZE command in order to be populated. If any field used in the expression is not populated, the calculated value returned for the expression is unpredictable.

Example: Using Prefix Operators With RECOMPUTE

The following request creates a calculated field named DIFF, which is the difference between DOLLARS and BUDDOLLARS. This value is then recomputed for each region, without using prefix operators:

```
TABLE FILE GGSALES
 SUM UNITS DOLLARS BUDDOLLARS
 COMPUTE DIFF/I10 = DOLLARS-BUDDOLLARS;
 BY REGION
 BY CATEGORY
 WHERE CATEGORY EQ 'Food' OR 'Coffee'
 WHERE REGION EQ 'West' OR 'Midwest'
 ON REGION RECOMPUTE
END
```

The recomputed value is the difference between the totals for DOLLARS and BUDDOLLARS:

Region	Category	Unit Sales	Dollar Sales	Budget Dollars	DIFF
Midwest	Coffee	332777	4178513	4086032	92481
	Food	341414	4338271	4220721	117550
*TOTAL Midwest		674191	8516784	8306753	210031
West	Coffee	356763	4473517	4523963	-50446
	Food	340234	4202337	4183244	19093
*TOTAL West		696997	8675854	8707207	-31353
TOTAL		1371188	17192638	17013960	178678

The following request uses prefix operators in the RECOMPUTE command to calculate maximum DOLLARS and the minimum BUDDOLLARS and then recompute DIFF. Regardless of which prefix operator you specify for DIFF, it is calculated as the difference between the values in the DOLLARS and BUDDOLLARS columns. If any of the fields used in the calculation (DOLLARS, BUDDOLLARS, and DIFF) do not display on the summary row, the calculation cannot be performed:

```
TABLE FILE GGSALES
 SUM UNITS DOLLARS BUDDOLLARS
 COMPUTE DIFF/I10 = DOLLARS-BUDDOLLARS;
 BY REGION
 BY CATEGORY
 WHERE CATEGORY EQ 'Food' OR 'Coffee'
 WHERE REGION EQ 'West' OR 'Midwest'
 ON REGION RECOMPUTE MAX. DOLLARS MIN. BUDDOLLARS AVE. DIFF
END
```

The output is:

Region	Category	Unit Sales	Dollar Sales	Budget Dollars	DIFF
Midwest	Coffee	332777	4178513	4086032	92481
	Food	341414	4338271	4220721	117550
*TOTAL Midwest			4338271	4086032	252239
West	Coffee	356763	4473517	4523963	-50446
	Food	340234	4202337	4183244	19093
*TOTAL West			4473517	4183244	290273
TOTAL			4473517	4086032	387485

Using Multiple SUB-TOTAL or SUMMARIZE Commands

Example:

Using Multiple SUB-TOTAL Commands With Prefix Operators

SUB-TOTAL and SUMMARIZE propagate their operations to all higher-level sort fields. If a request uses SUB-TOTAL or SUMMARIZE at multiple sort levels, more than one prefix operator may apply to the same field.

When a SUB-TOTAL or SUMMARIZE command on a lower-level sort field propagates up to the higher levels, it applies its prefix operators only to those fields that did not already have a different prefix operator specified at the higher level. For any field that had a prefix operator specified at a higher level, the original prefix operator is applied at the level at which it was first specified and on the grand total line, unless a different operator is specified for the grand total line.

Example: Using Multiple SUB-TOTAL Commands With Prefix Operators

```
DEFINE FILE GGSALES
 YEAR/YY = DATE;
END
TABLE FILE GGSALES
 SUM UNITS DOLLARS/D10.2 BUDDOLLARS
 BY YEAR
 BY ST
 BY REGION
 BY CATEGORY
 WHERE REGION EQ 'West' OR 'Midwest'
 WHERE ST EQ 'CA' OR 'IL'
 WHERE YEAR EQ '1996' OR '1997'
 ON YEAR SUB-TOTAL CNT. UNITS AS '*CNT. UNITS:'
 ON ST SUB-TOTAL AVE. DOLLARS AS '*AVE. $:'
 ON REGION SUB-TOTAL MIN. AS '*MIN.:'
END
```

In the following report output, some of the values have been manually italicized or bolded for clarity:

- The lines preceded by asterisks (*) and the very last (grand total) line are generated by the SUB-TOTAL commands.
- Subtotal values in the normal typeface are the count of unit sales generated by the command ON YEAR SUB-TOTAL CNT. UNITS. This command is the topmost summary command and therefore does not propagate to any other summary lines.
- Subtotal values in bold italic are average dollar sales generated by the command ON ST SUB-TOTAL AVE. DOLLARS. This command is the second summary command and therefore propagates to the DOLLARS column of summary lines for the YEAR sort field.

- Subtotal values in bold are minimums within their sort groups generated by the command ON REGION SUB-TOTAL MIN. This command is the last summary command and therefore propagates to all other summary lines, but only calculates minimum values for those columns not already populated with a count or an average.

YEAR	State	Region	Category	Unit Sales	Dollar Sales	Budget Dollars
1996	CA	West	Coffee	117539	1,484,873.00	1453548
			Food	116389	1,443,083.00	1414902
			Gifts	74948	947,783.00	946382
*MIN.: West				74948	**947,783.00**	946382
*AVE. $: CA				74948	***1,291,913.00***	946382
	IL	Midwest	Coffee	52348	683,559.00	628333
			Food	55867	729,658.00	705915
			Gifts	38405	481,515.00	487090
*MIN.: Midwest				38405	**481,515.00**	487090
*AVE. $: IL				38405	***631,577.33***	487090
*CNT. UNITS: 1996				6	961,745.17	487090
1997	CA	West	Coffee	118044	1,453,023.00	1507092
			Food	106322	1,325,430.00	1302582
			Gifts	77328	988,080.00	961841
*MIN.: West				77328	**988,080.00**	961841
*AVE. $: CA				77328	***1,255,511.00***	961841
	IL	Midwest	Coffee	57233	715,220.00	737931
			Food	59293	754,132.00	737912
			Gifts	37918	473,792.00	484611
*MIN.: Midwest				37918	**473,792.00**	484611
*AVE. $: IL				37918	***647,714.67***	484611
*CNT. UNITS: 1997				6	951,612.83	484611
TOTAL				12	956,679.00	484611

Handling Missing Data in HOLD Files

In this section:

Controlling Missing Values Propagated to a HOLD File

Holding Missing Values

Propagating Missing Values to a Variable Length Delimited HOLD File

Three SET parameters have been added to improve the handling of missing data in HOLD files:

- HNODATA controls how missing data characters are propagated to a HOLD file.
- HOLDMISS distinguishes between missing data and blank strings or zero values in a HOLD file.
- NULL enables you to create a variable length comma- or tab-delimited HOLD file that differentiates between a missing value and a blank string or zero value.

Controlling Missing Values Propagated to a HOLD File

How to:

Control Missing Values Propagated to a HOLD File

Example:

Controlling Missing Data Values in an Alphanumeric HOLD File

Propagating Numeric Values for Missing Data to a HOLD File

In Release 5.2.3, the SET HNODATA command controls the missing data characters that are propagated to fields with the MISSING = ON attribute in HOLD FORMAT ALPHA files. Missing values in fields that do not have the MISSING = ON attribute are propagated to a HOLD file as blank (for alphanumeric fields) or zero (for numeric fields).

Syntax: How to Control Missing Values Propagated to a HOLD File

```
SET HNODATA = charstring

ON TABLE SET HNODATA charstring
```

where:

charstring

Is a string of up to 12 characters that will be propagated to a HOLD FORMAT ALPHA file for missing values in a field with the MISSING = ON attribute. A period (.) is the default value.

If the string is longer than the length of the field, the value stored in:

- An alphanumeric field will be the leftmost characters of the string.
- A numeric field will be a blank string.

When an alphanumeric string other than the default value (the period) is used to populate a missing numeric field, a blank will be inserted in the held field to prevent a format error when displaying the data. If you use the default HNODATA value, it is inserted in numeric fields. In this way, a request against the HOLD file can recognize missing data that was propagated to the HOLD file.

If a number with decimal places is specified for HNODATA and the field with missing data requires an integer, the value is rounded to a whole number and then inserted. In a numeric field that supports decimal places, the value is rounded and inserted with the correct number of decimal digits.

Example: Controlling Missing Data Values in an Alphanumeric HOLD File

First, edit the BANK_NAME and BANK_CODE fields in the EMPLOYEE Master File to support missing data:

```
FIELDNAME = BANK_NAME, ALIAS = BN, FORMAT = A20, MISSING = ON ,$
FIELDNAME = BANK_CODE, ALIAS = BC, FORMAT = I6S, MISSING = ON ,$
```

The following request sets the HNODATA string to the string 'N/A' and creates an alphanumeric HOLD file named HMISS:

```
SET HNODATA = 'N/A'
TABLE FILE EMPLOYEE
 PRINT BANK_NAME BANK_CODE
 BY EMP_ID
 ON TABLE HOLD FORMAT ALPHA AS HMISS
END
-RUN
TABLE FILE HMISS
 PRINT *
END
```

Note that a blank is inserted in the integer field BANK_CODE:

EMP_ID	BANK_NAME	BANK_CODE
071382660	N/A	
112847612	N/A	
117593129	STATE	510271
119265415	N/A	
119329144	BEST BANK	302161
123764317	ASSOCIATED	112600
126724188	N/A	
219984371	N/A	
326179357	ASSOCIATED	112600
451123478	ASSOCIATED	112600
543729165	N/A	
818692173	BANK ASSOCIATION	175963

Example: Propagating Numeric Values for Missing Data to a HOLD File

First, edit the BANK_NAME and BANK_CODE fields in the EMPLOYEE Master File to support missing data:

```
FIELDNAME = BANK_NAME, ALIAS = BN, FORMAT = A20, MISSING = ON ,$
FIELDNAME = BANK_CODE, ALIAS = BC, FORMAT = I6S, MISSING = ON ,$
```

The request is:

```
SET HNODATA = 9999
TABLE FILE EMPLOYEE
 PRINT BANK_NAME BANK_CODE
 BY EMP_ID
 ON TABLE HOLD FORMAT ALPHA AS HMISS
END
-RUN
TABLE FILE HMISS
 PRINT *
END
```

The output is:

EMP_ID	BANK_NAME	BANK_CODE
071382660	9999	9999
112847612	9999	9999
117593129	STATE	510271
119265415	9999	9999
119329144	BEST BANK	302161
123764317	ASSOCIATED	112600
126724188	9999	9999
219984371	9999	9999
326179357	ASSOCIATED	112600
451123478	ASSOCIATED	112600
543729165	9999	9999
818692173	BANK ASSOCIATION	175963

Holding Missing Values

How to:

Store Missing Data in HOLD Files

Example:

Holding Missing Values Using HOLDMISS

In HOLD files, you can distinguish between missing data and default values of blank (for character data) or zero (for numeric data) using the SET HOLDMISS command.

Syntax: How to Store Missing Data in HOLD Files

```
SET HOLDMISS = {ON|OFF}

ON TABLE SET HOLDMISS {ON|OFF}
```

where:

`ON`

Allows you to store missing data in a HOLD file. When TABLE generates a default value for data not found, it generates missing values.

`OFF`

Does not allow you to store missing data in a HOLD file. OFF is the default value.

Example: Holding Missing Values Using HOLDMISS

```
SET HOLDMISS = ON
TABLE FILE MOVIES
 SUM WHOLESALEPR
 BY CATEGORY ACROSS RATING
 ON TABLE HOLD AS HLDM
END
-RUN
TABLE FILE HLDM
 PRINT *
END
```

The output is:

CATEGORY	WHOLESALEPR	WHOLESALEPR	WHOLESALEPR	WHOLESALEPR	WHOLESALEPR
ACTION	.	.	20.98	.	34.48
CHILDREN	54.49	51.38	.	.	.
CLASSIC	40.99	160.80	.	.	.
COMEDY	.	.	46.70	30.00	13.75
DRAMA	.	.	.	.	10.00
FOREIGN	13.25	.	62.00	.	70.99
MUSICALS	15.00	.	13.99	9.99	13.99
MYSTERY	.	9.00	18.00	9.00	80.97
SCI/FI	.	.	.	35.99	43.53
TRAIN/EX	.	60.98	.	.	.

Propagating Missing Values to a Variable Length Delimited HOLD File

How to:

Propagate Missing Values to a Delimited HOLD File

Example:

Creating a Comma-delimited File With Missing Values

In Release 5.2.3, you can create a variable-length comma- or tab-delimited HOLD file that differentiates between a missing value and a blank string or zero value using the SET NULL command.

Syntax: How to Propagate Missing Values to a Delimited HOLD File

```
SET NULL = {ON|OFF}
```

where:

ON

Propagates missing values to a delimited HOLD file when the field has MISSING = ON in the Master File. The HOLD formats supported for SET NULL = ON are COM, COMT, TAB, and TABT.

Missing values in a record are denoted by two consecutive delimiters. A record that starts with a missing value has a delimiter in the first position, and a record that ends with a missing value has a delimiter in the last position.

OFF

Propagates the value zero for a missing numeric value and blank ("") for a missing alphanumeric value to a delimited HOLD file. OFF is the default value.

Note:

- When SET NULL = ON, the setting for the HNODATA parameter is ignored.
- A missing date is indistinguishable from an empty date.

Example: Creating a Comma-delimited File With Missing Values

```
SET NULL = ON
TABLE FILE SALES
 PRINT RETURNS DAMAGED
 BY CITY BY DATE
 ON TABLE HOLD AS NULL1 FORMAT COMT
END
```

The NULL1 file contains the following records:

```
 "CITY","DATE","RETURNS","DAMAGED"

 "NEW YORK","1017",2,3

 "NEW YORK","1017",2,1

 "NEW YORK","1017",0,1              DAMAGED=0

 "NEW YORK","1017",,6               DAMAGED is missing

 "NEW YORK","1017",4,               RETURNS is missing

 "NEW YORK","1017",0,0              DAMAGED=0, RETURNS=0

 "NEW YORK","1017",3,2
```

```
"NEW YORK","1017",4,7
"NEW YORK","1017",,                DAMAGED is missing, RETURNS is
                                   missing
"NEW YORK","1017",4,2
"NEWARK","1018",1,1
"NEWARK","1019",1,0                RETURNS=0
```

Using Variable-Length Character Expressions

In this section:

Using AnV Fields With Functions

LENV: Returning the Length of an Alphanumeric Field

LOCASV: Creating a Variable Length Lowercase String

POSITV: Finding the Beginning of a Variable Length Substring

SUBSTV: Extracting a Variable Length Substring

TRIMV: Removing Characters From a String

UPCASV: Creating a Variable Length Uppercase String

How to:

Specify AnV Fields in a Master File

In Release 5.2.3, you can use an A*n*V field of alphanumeric type in arithmetic and logical expressions in the same way you would use an A*n* type field. An expression that contains A*n*V type fields can be of either the A*n*V or A*n* type. The type that results from the expression depends on the specific type of operation.

You can use the following with A*n*V fields:

- Concatenation
- EDIT function
- CONTAINS and OMITS operator
- LIKE operator
- EQ, NE, LT, GT, LE, and GE operators
- DECODE function
- Assignment operation (which assigns one value to equal another)

In addition, six functions have been designed to work specifically with the A*n*V data type:

- LENV
- LOCASV
- POSITV
- SUBSTV
- TRIMV
- UPCASV

For details and examples, see *Using AnV Fields With Functions* on page 1-52.

The character format A*n*V is supported in Master Files for FOCUS, XFOCUS, Fusion, and relational data sources. This format is used to represent the VARCHAR (variable length character) data types supported by relational database management systems.

For relational data sources, A*n*V keeps track of the actual length of a VARCHAR column. This information is especially valuable when the value is used to populate a VARCHAR column in a different RDBMS. It affects whether trailing blanks are retained in string concatenation and, for Oracle, string comparisons (the other relational engines ignore trailing blanks in string comparisons).

In a FOCUS, Fusion, or XFOCUS data source, A*n*V does not provide true variable length character support. It is a fixed-length character field with an extra two leading bytes to contain the actual length of the data stored in the field. This length is stored as a short integer value occupying two bytes. Because of the two bytes of overhead and the additional processing required to strip them, A*n*V format is *not* recommended for use with non-relational data sources.

Syntax: How to Specify AnV Fields in a Master File

```
FIELD = name, ALIAS = alias, USAGE = AnV [,ACTUAL = AnV] ,$
```

where:

n

Is the size (maximum length) of the field. It can be from 1 to 4093. Note that because of the additional two bytes used to store the length, an A4093V field is actually 4095 bytes long. A size of zero (A0V) is not supported. The length of an instance of the field can be zero.

Note: The HOLD FORMAT ALPHA command creates an ACTUAL format of A*n*V in the Master File.

Using AnV Fields With Functions

A*n*V fields can be used as arguments to all Information Builders-supplied functions that expect alphanumeric arguments. An A*n*V input parameter is treated as an A*n* parameter and is padded with blanks to its declared size (*n*). If the last parameter specifies an A*n*V format, the function result is converted to type A*n*V with actual length set equal to its size.

Note: Many functions require both an alphanumeric string and its length as input arguments. If the supplied string is stored in an A*n*V field, you still must supply a length argument to satisfy the requirements of the function's calling sequence. However, the length that will be used in the function's calculations is the actual length stored as the first two bytes of the A*n*V field.

The following functions are designed to work specifically with the A*n*V data type:

- LENV
- LOCASV
- POSITV
- SUBSTV
- TRIMV
- UPCASV

In general, any input argument can be a field or a literal. In most cases, numeric input arguments to these functions are supplied as literals, and there is no reason not to supply an integer value. However, if the value is not an integer, it is truncated to an integer value regardless of whether it was supplied as a field or a literal.

LENV: Returning the Length of an Alphanumeric Field

How to:

Find the Length of an Alphanumeric Field

Example:

Finding the Length of an AnV Field

The LENV function returns the actual length of an A*n*V input field or the size of an A*n* field.

Syntax: How to Find the Length of an Alphanumeric Field

```
LENV(string, outfield)
```

where:

string

Alphanumeric

Is the source field, or an alphanumeric constant enclosed in single quotation marks. If it is a field, it can have A*n* or A*n*V format. If it is a field of type A*n*V, its length is taken from the length bytes stored in the field.

outfield

Integer

Field that receives the actual length, or the format of the output value enclosed in single quotation marks.

Example: Finding the Length of an AnV Field

The TRIMV function creates an A*n*V field named TITLEV by removing trailing blanks from the TITLE value (see *TRIMV: Removing Characters From a String* on page 1-61). The LENV function returns the actual length of each instance of TITLEV to the ALEN field:

```
TABLE FILE MOVIES
 PRINT
 COMPUTE TITLEV/A39V = TRIMV('T', TITLE, 39, ' ', 1, TITLEV);
 COMPUTE ALEN/I2 = LENV(TITLEV, ALEN);
 BY CATEGORY NOPRINT
 WHERE CATEGORY EQ 'CHILDREN'
END
```

The output is:

TITLEV	ALEN
SMURFS, THE	11
SHAGGY DOG, THE	15
SCOOBY-DOO-A DOG IN THE RUFF	28
ALICE IN WONDERLAND	19
SESAME STREET-BEDTIME STORIES AND SONGS	39
ROMPER ROOM-ASK MISS MOLLY	26
SLEEPING BEAUTY	15
BAMBI	5

LOCASV: Creating a Variable Length Lowercase String

How to:

Create a Variable Length Lowercase String

Example:

Creating a Variable Length Lowercase String

The LOCASV function converts alphabetic characters to lowercase and is similar to LOCASE. However, LOCASV can return A*n*V output whose actual length is the lesser of the actual length of the A*n*V input field and an input parameter that specifies the length limit.

Syntax: How to Create a Variable Length Lowercase String

```
LOCASV(length_limit, string, outfield)
```

where:

length_limit

Numeric

Is a positive constant or a field whose integer portion represents the size and, therefore, the upper limit for the length of the input string.

string

Alphanumeric

Is the field, or alphanumeric constant enclosed in single quotation marks, that is to be converted to lowercase. If it is a field, it can have A*n* or A*n*V format. If it is a field of type A*n*V, its length is taken from the length bytes stored in the field. If *length_limit* is smaller than the actual length, the source string is truncated to this upper limit.

outfield

Alphanumeric

Is either a field with A*n*V or A*n* format for the returned lowercase string, or the format of the output value enclosed in single quotation marks.

If the format of *outfield* is A*n*V, the actual length returned is equal to the smaller of the input string length and *length_limit*.

Example: Creating a Variable Length Lowercase String

The LOCASV function converts the LAST_NAME field to lowercase and specifies a length limit of five characters. The results are stored in the LOWCV_NAME field:

```
TABLE FILE EMPLOYEE
 PRINT LAST_NAME AND COMPUTE
 LOWCV_NAME/A15 = LOCASV(5, LAST_NAME, LOWCV_NAME);
 WHERE DEPARTMENT EQ 'MIS';
END
```

The output is:

LAST_NAME	LOWCV_NAME
SMITH	smith
JONES	jones
MCCOY	mccoy
BLACKWOOD	black
GREENSPAN	green
CROSS	cross

POSITV: Finding the Beginning of a Variable Length Substring

How to:

Find the Beginning of a Variable Length Substring

Example:

Finding the Starting Position of a Variable Length Pattern

The POSITV function is similar to POSIT. However, the lengths of its A*n*V parameters are based on the actual lengths of those parameters in comparison with two other parameters that specify their sizes.

Syntax: How to Find the Beginning of a Variable Length Substring

```
POSITV(parent, in_limit, substring, sub_limit, outfield)
```

where:

parent

Alphanumeric

Is the field, or alphanumeric constant enclosed in single quotation marks, that contains the substring whose position you want to find. If it is a field, it can have A*n* or A*n*V format. If it is a field of type A*n*V, its length is taken from the length bytes stored in the field. If *in_limit* is smaller than the actual length, the source string is truncated to this upper limit.

in_limit

Numeric

Is a positive constant or a field whose integer portion represents the size and, therefore, the upper limit for the length of the input string.

substring

Alphanumeric

Is a field name, or an alphanumeric constant enclosed in single quotation marks, that contains the substring whose position you want to locate. If it is a field, it can have A*n* or A*n*V format. If it is a field of type A*n*V, its length is taken from the length bytes stored in the field. If *sub_limit* is smaller than the actual length, the source string is truncated to this upper limit.

sub_limit

Numeric

Is a positive constant of a field whose integer portion represents the size and, therefore, the upper limit for the length of the substring.

outfield

Integer

Is the name of the field to which the position is returned, or the format of the output value enclosed in single quotation marks.

Example: Finding the Starting Position of a Variable Length Pattern

The POSITV function finds the starting position of a trailing definite or indefinite article in a movie title (such as ", THE" in SMURFS, THE). First, TRIMV removes the trailing blanks from the title so that the article will be the trailing pattern (see *TRIMV: Removing Characters From a String* on page 1-61):

```
DEFINE FILE MOVIES
 TITLEV/A39V = TRIMV('T', TITLE, 39, ' ', 1, TITLEV);
 PSTART/I4 = POSITV(TITLEV,LENV(TITLEV, 'I4'), ',', 1, 'I4');
 PLEN/I4 = IF PSTART NE 0 THEN LENV(TITLEV, 'I4') - PSTART +1 ELSE 0;
END
TABLE FILE MOVIES
 PRINT TITLE
 PSTART AS 'Pattern,Start' IN 25
 PLEN AS 'Pattern,Length'
 BY CATEGORY NOPRINT
 WHERE PLEN NE 0
END
```

The output is:

TITLE	Pattern Start	Pattern Length
SMURFS, THE	7	5
SHAGGY DOG, THE	11	5
MALTESE FALCON, THE	15	5
PHILADELPHIA STORY, THE	19	5
TIN DRUM, THE	9	5
FAMILY, THE	7	5
CHORUS LINE, A	12	3
MORNING AFTER, THE	14	5
BIRDS, THE	6	5
BOY AND HIS DOG, A	16	3

SUBSTV: Extracting a Variable Length Substring

How to:

Extract a Variable Length Substring

Example:

Extracting a Variable Length Substring

The SUBSTV function extracts a substring from a string and is similar to SUBSTR. However, the end position for the string is calculated from the starting position and the substring length. Therefore, it has fewer parameters than SUBSTR. Also, the actual length of the output field, if it is an A*n*V field, is determined based on the substring length.

Syntax: How to Extract a Variable Length Substring

```
SUBSTV(in_limit, parent, start, sublength, outfield)
```

where:

in_limit

Numeric

Is a positive constant or a field whose integer portion represents the size and, therefore, the upper limit for the length of the input string.

parent

Alphanumeric

Is the field, or alphanumeric constant enclosed in single quotation marks, that contains the substring you want to extract. If it is a field, it can have A*n* or A*n*V format. If it is a field of type A*n*V, its length is taken from the length bytes stored in the field. If *in_limit* is smaller than the actual length, the source string is truncated to this size. The final length value determined by this comparison will be referred to as *p_length* (see the description of the *outfield* parameter).

start

Numeric

Is a positive constant or a field whose integer portion represents the starting position of the substring. Note that the starting position can exceed the input string length.

sublength

Numeric

Is a positive constant or a field whose integer portion represents the length of the substring. The end position of the substring is *end* =*start* + *sublength* -1. Note that the ending position can exceed the input string length depending on the provided values for *start* and *sublength* provided.

outfield

Alphanumeric

Is either a field with A*n*V or A*n* format for the returned substring, or the format of the output value enclosed in single quotation marks.

If the format of *outfield* is A*n*V, the actual length, *outlen*, is computed as follows from the values for *end*, *start*, and *p_length* (see the *parent* parameter):

If *end* > *p_length* or *end* < *start*, then *outlen* = 0 otherwise, *outlen* = *end* - *start* + 1.

Example: Extracting a Variable Length Substring

The following request extracts a trailing definite or indefinite article from a movie title (such as ", THE" in "SMURFS, THE"). First, it trims the trailing blanks so that the article is the trailing pattern. Next, it finds the starting position and length of the pattern. Then, SUBSTV extracts the pattern and TRIMV trims the pattern from the title. (See *POSITV: Finding the Beginning of a Variable Length Substring* on page 1-56 and *TRIMV: Removing Characters From a String* on page 1-61):

```
DEFINE FILE MOVIES
 TITLEV/A39V = TRIMV('T', TITLE, 39, ' ', 1, TITLEV);
 PSTART/I4 = POSITV(TITLEV,LENV(TITLEV, 'I4'), ',', 1,'I4');
 PLEN/I4 = IF PSTART NE 0 THEN LENV(TITLEV, 'I4') - PSTART +1  ELSE 0;
 PATTERN/A20V = SUBSTV(39, TITLEV, PSTART, PLEN, PATTERN);
 NEWTIT/A39V = TRIMV('T',TITLEV,39,PATTERN,LENV(PATTERN,'I4'), NEWTIT);
END
TABLE FILE MOVIES
 PRINT TITLE
 PSTART AS 'Pattern,Start' IN 25
 PLEN AS 'Pattern,Length'
 NEWTIT AS 'Trimmed,Title' IN 55
 BY CATEGORY NOPRINT
 WHERE PLEN NE 0
END
```

The output is:

TITLE	Pattern Start	Pattern Length	Trimmed Title
SMURFS, THE	7	5	SMURFS
SHAGGY DOG, THE	11	5	SHAGGY DOG
MALTESE FALCON, THE	15	5	MALTESE FALCON
PHILADELPHIA STORY, THE	19	5	PHILADELPHIA STORY
TIN DRUM, THE	9	5	TIN DRUM
FAMILY, THE	7	5	FAMILY
CHORUS LINE, A	12	3	CHORUS LINE
MORNING AFTER, THE	14	5	MORNING AFTER
BIRDS, THE	6	5	BIRDS
BOY AND HIS DOG, A	16	3	BOY AND HIS DOG

TRIMV: Removing Characters From a String

How to:

Remove Characters From a String

Example:

Creating an AnV Field by Removing Trailing Blanks

The TRIMV function removes a pattern from a string and is similar to TRIM. However, both the input string and the pattern of characters to remove can have A*n*V format.

Note: The TRIMV function is useful for converting an A*n* field to an A*n*V field (with the length bytes containing the actual length of the data up to the last non-blank character).

Syntax: How to Remove Characters From a String

```
TRIMV(trim_where, string, slength_limit, pattern, plength_limit, outfield)
```

where:

trim_where

Alphanumeric

Is one of the following, which indicates where to remove the pattern:

'L' removes leading occurrences.

'T' removes trailing occurrences.

'B' removes both leading and trailing occurrences.

string

Alphanumeric

Is the source field, or an alphanumeric constant enclosed in single quotation marks. If it is a field, it can have A*n* or A*n*V format. If it is a field of type A*n*V, its length is taken from the length bytes stored in the field. If *slength_limit* is smaller than the actual length, the source string is truncated to this upper limit.

slength_limit

Numeric

Is a positive numeric value. The integer part is an upper limit for the length of the input string.

pattern

Alphanumeric

Is a field, or an alphanumeric constant enclosed in single quotation marks, that is the pattern to remove. If it is a field, it can have A*n* or A*n*V format. If it is a field of type A*n*V, its length is taken from the length bytes stored in the field. If *plength_limit* is smaller than the actual length, the pattern is truncated to this limit.

plength_limit

Numeric

Is a positive numeric value. The integer part is an upper limit for the length of the pattern.

outfield

Alphanumeric

Is either a field with A*n*V or A*n* format for the returned substring, or the format of the output value enclosed in single quotation marks.

If the format of *outfield* is A*n*V, the actual length is equal to the number of characters left after trimming.

Example: Creating an AnV Field by Removing Trailing Blanks

The TRIMV function creates an A*n*V field named TITLEV by removing trailing blanks from the TITLE value:

```
TABLE FILE MOVIES
 PRINT DIRECTOR
 COMPUTE TITLEV/A39V = TRIMV('T', TITLE, 39, ' ', 1, TITLEV);
 BY CATEGORY
END
```

Here are the first 10 lines of the output:

CATEGORY	DIRECTOR	TITLEV
ACTION	SPIELBERG S.	JAWS
	VERHOVEN P.	ROBOCOP
	VERHOVEN P.	TOTAL RECALL
	SCOTT T.	TOP GUN
	MCDONALD P.	RAMBO III
CHILDREN		SMURFS, THE
	BARTON C.	SHAGGY DOG, THE
		SCOOBY-DOO-A DOG IN THE RUFF
	GEROMINI	ALICE IN WONDERLAND
		SESAME STREET-BEDTIME STORIES AND SONGS

UPCASV: Creating a Variable Length Uppercase String

How to:

Create a Variable Length Uppercase String

Example:

Creating a Variable Length Uppercase String

The UPCASV function converts alphabetic characters to uppercase and is similar to UPCASE. However, UPCASV can return A*n*V output whose actual length is the lesser of the actual length of the A*n*V input field and an input parameter that specifies the size.

Syntax: How to Create a Variable Length Uppercase String

```
UPCASV(length_limit, string, outfield)
```

where:

length_limit

Numeric

Is a positive constant or a field whose integer portion represents the size and, therefore, the upper limit for the length of the input string.

string

Alphanumeric

Is the field, or alphanumeric constant enclosed in single quotation marks, that will be converted to uppercase. If it is a field, it can have A*n* or A*n*V format. If it is a field of type A*n*V, its length is taken from the length bytes stored in the field. If *length_limit* is smaller than the actual length, the source string is truncated to this size.

outfield

Alphanumeric

Is either a field with A*n*V or A*n* format for the returned uppercase string, or the format of the output value enclosed in single quotation marks.

If the format of *outfield* is A*n*V, then the actual length returned is equal to the smaller of the input string length and *length_limit*.

Example: Creating a Variable Length Uppercase String

Suppose you are sorting on a field that contains both uppercase and mixed-case values. The following request defines a field called LAST_NAME_MIXED that contains both uppercase and mixed-case values:

```
DEFINE FILE EMPLOYEE
 LAST_NAME_MIXED/A15 = IF DEPARTMENT EQ 'MIS' THEN LAST_NAME ELSE
  LCWORD(15, LAST_NAME, 'A15');
 LAST_NAME_UPCASV/A15 = UPCASV(5, LAST_NAME_MIXED, 'A15') ;
END
```

Suppose you execute a request that sorts by this field:

```
TABLE FILE EMPLOYEE
 PRINT LAST_NAME_MIXED AND FIRST_NAME BY LAST_NAME_UPCASV
 WHERE CURR_JOBCODE EQ 'B02' OR 'A17' OR 'B04';
END
-RUN
```

The output is:

LAST_NAME_UPCASV	LAST_NAME_MIXED	FIRST_NAME
BANNI	Banning	JOHN
BLACK	BLACKWOOD	ROSEMARIE
CROSS	CROSS	BARBARA
MCCOY	MCCOY	JOHN
MCKNI	Mcknight	ROGER
ROMAN	Romans	ANTHONY

Validating Function Arguments

How to:

Enable Function Argument Verification

Control Function Argument Verification

Example:

Verifying Arguments With Correctable Errors

Verifying Arguments With Uncorrectable Errors

The SET USERFCHK command controls the level of verification applied to DEFINE FUNCTION arguments and Information Builders-supplied function arguments. USERFCHK does not affect verification of the number of arguments; the correct number must always be supplied.

Functions typically expect arguments to be a specific type or have a length that depends on the value of another argument. It is possible in some situations to enforce these rules by truncating the length of an argument and, therefore, avoid generating an error at run time.

The level of verification and conversion performed depends on the specific function. The following two situations can usually be fixed satisfactorily:

- If a numeric argument specifies a maximum size for an alphanumeric argument, but the alphanumeric string supplied is longer than the specified size, the string can be truncated.
- If an argument supplied as a numeric literal specifies a value larger than the maximum size for an argument, it can be reduced to the proper size.

Since argument verification can only be used for functions supplied by Information Builders, you must first ensure that those functions are enabled using the SET USERFNS command.

Syntax: How to Enable Function Argument Verification

If your site has a locally written function with the same name as an Information Builders-supplied function, the setting of the USERFNS parameter determines which function will be used.

```
SET USERFNS = {SYSTEM|LOCAL}
```

where:

SYSTEM

Gives precedence to functions supplied by Information Builders and those created by the DEFINE FUNCTION command. This setting is required to enable argument verification. SYSTEM is the default value.

LOCAL

Gives precedence to locally written functions. Argument verification is not performed when this setting is in effect.

Syntax: How to Control Function Argument Verification

Note that the USERFNS = SYSTEM setting must be in effect when you issue the following command in a profile or procedure. See *Enable Function Argument Verification* on page 1-67.

```
SET USERFCHK = setting
```

where:

setting

Can be one of the following:

ON is the default value. This value verifies parameters in requests, but does not verify parameters for functions used in Master File DEFINEs. If a parameter has an incorrect length, an attempt is made to fix the problem. If such a problem cannot be fixed, a message is generated and the evaluation of the affected expression is terminated.

Because parameters are not verified for functions specified in a Master File, no errors are reported for those functions until the DEFINE field is used in a subsequent request. If a problem occurs, the following message is generated:

```
(FOC003) THE FIELDNAME IS NOT RECOGNIZED
```

OFF does not verify parameters except in the following cases:

- If a parameter that is too long would overwrite the memory area in which the computational code is stored, the size is automatically reduced without a message being issued.
- If an alphanumeric parameter is too short, it is padded with blanks to the correct length.

`FULL` is the same as ON, but also verifies parameters for functions used in Master File DEFINEs.

`ALERT` verifies parameters in a request without halting execution when a problem is detected. It does not verify parameters for functions used in Master File DEFINEs. If a parameter has an incorrect length and an attempt is made to fix the problem behind the scenes, the problem is corrected with no message. If such a problem cannot be fixed, a warning message is generated. Execution then continues as though the setting were OFF.

Note: For the ON, FULL, and ALERT settings, if an argument provided is the incorrect type, verification fails and processing terminates.

Example: Verifying Arguments With Correctable Errors

The following request uses the SUBSTR function to extract the substring that starts in position 6 and ends in position 14 of the TITLE field. The fifth argument specifies a substring length (0) that is too short (it should be 9):

```
SET USERFCHK = ON
TABLE FILE MOVIES
 PRINT TITLE
 COMPUTE NEWTITLE/A9 = SUBSTR(39, TITLE, 6 ,14,0, NEWTITLE);
 WHERE CATEGORY EQ 'CHILDREN'
END
```

- When the request is executed with the USERFCHK = ON setting, a warning message is produced. The incorrect length is corrected and the request completes processing:

  ```
  (FOC36335) PARAMETER LENGTH CONFLICT IN FUNCTION "SUBSTR", ARG 5.
  ```

  The output is:

  ```
  TITLE
  SMURFS, THE
  SHAGGY DOG, THE
  SCOOBY-DOO-A DOG IN THE RUFF
  ALICE IN WONDERLAND
  SESAME STREET-BEDTIME STORIES AND SONGS
  ROMPER ROOM-ASK MISS MOLLY
  SLEEPING BEAUTY
  BAMBI
  ```

  Note: To view the message, right-click the report and select *View Source*.

- When the request is executed with the USERFCHK = OFF setting, no warning message is produced. The length error is still corrected and processing continues.

Example: Verifying Arguments With Uncorrectable Errors

The following request has an incorrect data type in the last argument to SUBSTR. This argument should specify an alphanumeric field or format for the extracted substring:

```
SET USERFCHK = ON
TABLE FILE MOVIES
 PRINT TITLE
 COMPUTE NEWTITLE/F9 = SUBSTR(39, TITLE, 6 ,14, 500, 'F9');
 WHERE CATEGORY EQ 'CHILDREN'
END
```

- When the request is executed with the USERFCHK = ON setting, a message is produced and the request terminates:

  ```
  ERROR AT OR NEAR LINE 5 IN PROCEDURE USERFC3 FOCEXEC
  (FOC279) NUMERIC ARGUMENTS IN PLACE WHERE ALPHA ARE CALLED FOR
  (FOC009) INCOMPLETE REQUEST STATEMENT
  UNKNOWN FOCUS COMMAND WHERE
   BYPASSING TO END OF COMMAND
  ```

- When the request is executed with the USERFCHK = OFF setting, no verification is done and no message is produced. The request runs and produces incorrect results. In some environments, this type of error may cause abnormal termination of the application:

TITLE	NEWTITLE
SMURFS, THE	*********
SHAGGY DOG, THE	*********
SCOOBY-DOO-A DOG IN THE RUFF	*********
ALICE IN WONDERLAND	0
SESAME STREET-BEDTIME STORIES AND SONGS	*********
ROMPER ROOM-ASK MISS MOLLY	*********
SLEEPING BEAUTY	0
BAMBI	0

Financial Reporting Language Enhancements

In this section:

Defining an FML Hierarchy

Displaying a Dynamic FML Hierarchy

Consolidating Data in an FML Hierarchy

Indenting Row Titles in an FML Hierarchy

Returning FOR Field Values in a Computed Column

Formatting an FML Report

You can define hierarchical relationships between fields in a Master File and automatically display the hierarchy using the Financial Modeling Language (FML).

For example, suppose that:

- An employee data source contains both the employee's ID and the ID of the employee's manager.

  or

- A general ledger data source contains both an account number field and an account parent field.

By examining these fields, it is possible to construct the entire organization chart or chart of accounts.

Once you have defined the hierarchy, it is loaded into memory. An FML request can then dynamically construct the rows that represent the hierarchical relationship and display them in the report starting at any point in the hierarchy.

For details about the process through which you define and use the FML hierarchy, see:

- *Defining an FML Hierarchy* on page 1-71.
- *Displaying a Dynamic FML Hierarchy* on page 1-73.
- *Consolidating Data in an FML Hierarchy* on page 1-78.
- *Returning FOR Field Values in a Computed Column* on page 1-87.
- *Formatting an FML Report* on page 1-89. This topic describes a feature introduced in Release 5.2.3, which enables you to control the indentation of rows and labels.

For the graphical implementation of the hierarchy features, see *Financial Report Painter Enhancements* in Chapter 2, *Developer Studio Enhancements*.

The FML language and the graphical Financial Report Painter are fully documented in the *Creating Financial Reports* manual.

Defining an FML Hierarchy

How to:

Define a Hierarchy Between Fields in a Master File

Assign Descriptive Captions for Hierarchy Field Values in the Master File

You define the hierarchical relationship between two fields in the Master File using the PROPERTY = PARENT_OF and REFERENCE = *hierachyfield* attributes.

The parent and child fields must share data values, and their relationship should be hierarchical. The formats of the parent and child fields must both be numeric or both alphanumeric.

You can also provide descriptive captions in the Master File for display in reports in place of the specified hierarchy field values.

Syntax: How to Define a Hierarchy Between Fields in a Master File

```
FIELD=parentfield,..., PROPERTY=PARENT_OF, REFERENCE=[seg.]hierarchyfield ,$
```

where:

parentfield

Is the parent field in the hierarchy.

Note: The parent and child fields in the hierarchy must share data values. The formats of the parent and child fields must both be numeric or both alphanumeric.

PROPERTY=PARENT_OF

Identifies this field as the parent of the referenced field in a hierarchy.

These attributes can be specified on every field. Therefore, multiple hierarchies can be defined in one Master File. However, an individual field can have only one parent. If multiple fields have PARENT_OF attributes for the same hierarchy field, the first parent found by traversing the structure in top-down, left-to-right order is used as the parent.

seg

Is the segment location of the hierarchy field. Required if more than one segment has a field named *hierarchyfield*.

hierarchyfield

Is the child field in the hierarchy.

PARENT_OF is also allowed on a virtual field in the Master File:

```
DEFINE name/fmt=expression;,PROPERTY=PARENT_OF, REFERENCE=hierarchyfield ,$
```

Syntax: How to Assign Descriptive Captions for Hierarchy Field Values in the Master File

To make reports generated from this data more meaningful, you can specify a caption for a hierarchy field in a Master File

```
FIELD=captionfield, ..., PROPERTY=CAPTION, REFERENCE=[seg.]hierarchyfield ,$
```

where:

captionfield

Is the name of the field that contains the descriptive text for the hierarchy field. For example, if the employee ID is the hierarchy field, the last name may be the descriptive text to be displayed on the report in place of the ID.

PROPERTY=CAPTION

Signifies that this field contains a descriptive caption to be displayed in place of the hierarchy field values.

A caption can be specified for every field, but an individual field can have only one caption. If multiple fields have CAPTION attributes for the same hierarchy field, the first parent found by traversing the structure in top-down, left-to-right order will be used as the caption.

seg

Is the segment location of the hierarchy field. Required if more than one segment has a field named *hierarchyfield*.

hierarchyfield

Is the hierarchy field.

CAPTION is also allowed on a virtual field in the Master File:

```
DEFINE name/fmt=expression;,PROPERTY=CAPTION, REFERENCE=hierarchyfield ,$
```

Displaying a Dynamic FML Hierarchy

How to:

Display an FML Hierarchy

Example:

Displaying an FML Hierarchy

Displaying an FML Hierarchy With Captions

The GET CHILDREN and WITH CHILDREN commands dynamically retrieve and display hierarchical data on the FML report. GET CHILDREN displays only the children, not the parent value referenced in the command. WITH CHILDREN displays the parent and then the children.

When a request references a hierarchy, an internal TABLE request is automatically run to produce the list of parent values and their associated children:

```
TABLE FILE chartfile
 BY parentfield BY hierarchyfield
 [SUM captionfield]
END
```

The resulting chart contains the following information. It may also contain the associated captions, depending on whether the AS CAPTION phrase is used in the request:

```
parentfield     hierarchyfield
-----------     --------------
parentvalue1    child1
parentvalue1    child2
parentvalue1    child3
   .
   .
   .
```

Syntax: How to Display an FML Hierarchy

```
TABLE FILE filename
 {PRINT|SUM} ...
 FOR hierarchyfield
 parentvalue {GET|WITH} CHILD[REN] [n|ALL] [AS CAPTION|'text'] [LABEL label]
.
.
.
END
```

where:

filename

Is the name of the file to be used in the FML request. (If the hierarchy for this request cannot be loaded automatically, you must have previously issued the LOAD CHART command to load the hierarchy.)

hierarchyfield

Is the hierarchy field name. If the request references a joined structure, the name must be the field name from the host file. The alias name is not supported.

parentvalue

Is the parent value for which the children are to be retrieved.

GET CHILDREN

Displays the hierarchy starting from the first child of the specified *parentvalue*. Does not include the parent in the display. (This corresponds to the FML syntax CHILD1 OVER CHILD2 OVER ...).

WITH CHILDREN

Displays the hierarchy starting from the specified *parentvalue*. Includes the parent in the display. (This corresponds to the FML syntax *parentvalue* OVER CHILD1 OVER CHILD2 OVER ...).

n|ALL

Is a positive integer from 1 to 99 specifying the number of levels of the hierarchy to display. If a number greater than 99 is specified, a warning message is displayed and *n* is set to 99. 1 is the default value. Therefore, if *n* is omitted, direct children are displayed. GET or WITH CHILDREN 2 displays direct children and grandchildren. GET or WITH CHILDREN 99 displays children to 99 levels. ALL is a synonym for 99. Each child instance is printed over the next child instance. Successive levels of the hierarchy field are indented two spaces from the previous level.

`CAPTION`

Indicates that caption values should be taken from the field defined as the CAPTION in the Master File.

Note: The AS CAPTION phrase is supported for tagged rows, including those that do not use the GET/WITH CHILDREN or ADD syntax. However, the hierarchy must be defined (by specifying the PARENT_OF attribute) in order to load and display the caption values. If the hierarchy is not defined, the AS CAPTION phrase is ignored.

`'text'`

Is a text string to use as the row title for the hierarchy field values. The CAPTION field defined in the Master File will not be used as the caption on the report output.

label

Is an explicit row label. Each generated row is labeled with the specified label text.

The hierarchy is displayed sorted by the parent field and, within each parent, sorted by the hierarchy field.

Note: In order to display the hierarchy with levels of data indented, you must include the SET BLANKINDENT command in your request. For detailed syntax, see *Indenting Row Titles in an FML Hierarchy* on page 1-85.

Example: Displaying an FML Hierarchy

The following request displays two levels of account numbers, starting from account 3000:

```
SET BLANKINDENT = ON
TABLE FILE CENTGL
 PRINT GL_ACCOUNT_PARENT
 FOR GL_ACCOUNT
 3000 WITH CHILDREN 2
END
```

The output is:

```
             Parent
3000         1000
    3100     3000
        3110 3100
        3120 3100
        3130 3100
        3140 3100
    3200     3000
        3300 3200
        3400 3200
        3500 3200
        3600 3200
        3700 3200
        3800 3200
        3900 3200
```

Note that if the request specifies GET CHILDREN instead of WITH CHILDREN, the line for the parent value (3000) does not display on the report output.

Example: Displaying an FML Hierarchy With Captions

The following request displays two levels of hierarchy starting with account 1000 (the top of the hierarchy) and displays the caption field values instead of the account numbers:

```
SET BLANKINDENT = ON
TABLE FILE CENTGL
 PRINT GL_ACCOUNT_PARENT
 FOR GL_ACCOUNT
 1000 WITH CHILDREN 2 AS CAPTION
END
```

The output is:

	Parent
Profit Before Tax	
Gross Margin	1000
Sales Revenue	2000
Cost Of Goods Sold	2000
Total Operating Expenses	1000
Selling Expenses	3000
General + Admin Expenses	3000
Total R+D Costs	1000
Salaries	5000
Misc. Equipment	5000

Note that if the request had specified GET CHILDREN instead of WITH CHILDREN, the line for the parent value (1000, Profit Before Tax) would not have displayed on the report output.

Consolidating Data in an FML Hierarchy

How to:

Create One Summary Row for an FML Hierarchy

Consolidate FML Hierarchy Data to Any Level and Depth

Display the Same Record in Multiple FOR Rows

Example:

Displaying One Summary Row for an FML Hierarchy

Displaying Consolidated Data For Multiple Hierarchy Levels

The ADD command consolidates multiple levels of the hierarchy on one line of the FML report output. ADD can be used alone or in conjunction with GET CHILDREN or WITH CHILDREN. Note that ADD is designed to work with requests that use the SUM command. It is also designed to be used with detail level data, not data that is consolidated.

When used alone, ADD aggregates the parent and children on one line of the report output, summing the numeric data values included on the line. (This corresponds to the FML syntax *parentvalue* or CHILD1 OR CHILD2 OR ...).

When used in conjunction with GET CHILDREN, ADD displays one line for each child of the specified parent value. Each line is a summation of that child and all of its children. You can specify the number of levels of children to display (which determines the number of lines generated on the report output) and the depth of summation under each child. By default, only direct children will have a line in the report output and the summary for each child will include all of its children.

When used in conjunction with WITH CHILDREN, ADD first displays a line in the report output that consists of the summation of the parent value and all of its children. Then, it displays additional lines identical to those displayed by GET CHILDREN ADD.

Syntax: How to Create One Summary Row for an FML Hierarchy

```
TABLE FILE filename
 SUM ...
 FOR hierarchyfield
 parentvalue ADD [n|ALL] [AS CAPTION|'text'] [LABEL label]
  .
  .
  .
END
```

where:

filename

Is the name of the file to be used in the FML request. (If the hierarchy for this request cannot be loaded automatically, you must have previously issued the LOAD CHART command to load the hierarchy.)

hierarchyfield

Is the hierarchy field name. If the request references a joined structure, the name must be the field name from the host file. The alias name is not supported.

parentvalue

Is the parent value that determines the starting point in the hierarchy for the aggregation.

ADD

Displays the parent and *n* levels of its children on one row, summing the numeric data values displayed on the row. (This corresponds to the FML syntax *parentvalue* or CHILD1 OR CHILD2 OR ...).

To display the sum of just the children, you must display the parent row, display the summary row, and use a recap to subtract the parent row from the sum. For example:

```
FOR ...
parentvalue                 OVER
parentvalue ADD 1           OVER
RECAP CHILDSUM = R2-R1;
```

n|ALL

Is a positive integer from 1 to 99 specifying the number of levels of the hierarchy to aggregate. ALL is the default value. Therefore, if *n* is omitted, all children are included in the sum. If *n* is 1, direct children are included. If *n* is 2, direct children and grandchildren are included. ADD 99 includes children to 99 levels. ALL is a synonym for 99.

`CAPTION`

Indicates that the caption of the parent value will display for the total row.

Note: The AS CAPTION phrase is supported for tagged rows, including those that do not use the GET/WITH CHILDREN or ADD syntax. However, the hierarchy must be defined (by specifying the PARENT_OF attribute) in order to load and display the caption values. If the hierarchy is not defined, the AS CAPTION phrase is ignored.

'text'

Is a text string to use as the row title for the aggregate row. The CAPTION field defined in the Master File will not be used as the caption on the report output.

label

Is an explicit row label. Each generated row is labeled with the specified label text.

Note: Data consolidation requires the reuse of FOR field values in multiple rows of an FML report. This capability is enabled by the SET FORMULTIPLE command, which must be included in the FML request. For detailed syntax, see *Display the Same Record in Multiple FOR Rows* on page 1-85.

Example: Displaying One Summary Row for an FML Hierarchy

The CENTSYSF data source contains detail level financial data. To use the account hierarchy in the CENTGL data source with this financial data, the two data sources are joined.

First, the lines of the hierarchy starting with account 3100 (Selling Expenses) are displayed by the WITH CHILDREN command. Note that only accounts with no children are populated in this detail level data source. The ADD command then creates one line that is the sum of account 3100 and all of its children:

```
SET BLANKINDENT = ON
SET FORMULTIPLE = ON
JOIN SYS_ACCOUNT IN CENTGL TO ALL SYS_ACCOUNT IN CENTSYSF
TABLE FILE CENTGL
 SUM NAT_AMOUNT/D10.0 NAT_YTDAMT/D10.0
 FOR GL_ACCOUNT
 3100 WITH CHILDREN ALL AS CAPTION OVER
 BAR                                OVER
 3100 ADD AS CAPTION
 IF PERIOD EQ '2002/03'
END
```

The output is:

	Month Actual	YTD Actual
Selling Expenses	.	.
Advertising	.	.
TV/Radio	1,049,146.	2,954,342.
Print Media	244,589.	721,448.
Internet Advertising	9,542.	29,578.
Promotional Expenses	53,719.	151,732.
Joint Marketing	97,135.	289,799.
Bonuses/Commisions	100,188.	304,199.
Selling Expenses	1,554,319.	4,451,098.

Syntax: How to Consolidate FML Hierarchy Data to Any Level and Depth

```
TABLE FILE filename
 SUM ...
 FOR hierarchyfield
 parentvalue {GET|WITH} CHILD[REN] [n|ALL] ADD [m|ALL]
  [AS CAPTION|'text'] [LABEL label]
 .
 .
 .
END
```

where:

filename

Is the name of the file to be used in the FML request. (If the hierarchy for this request cannot be loaded automatically, you must have previously issued the LOAD CHART command to load the hierarchy.)

hierarchyfield

Is the hierarchy field name. If the request references a joined structure, the name must be the field name from the host file. The alias name is not supported.

parentvalue

Is the parent value that determines the starting point in the hierarchy for the aggregation.

`GET|WITH CHILDREN`

GET specifies that the first line generated on the report is the consolidated line for the first child of the parent value. WITH specifies that the first line generated on the report is the consolidated line for the parent value, followed by the consolidated lines for each of its children to the level specified by *n*.

n|ALL

Is a positive integer from 1 to 99 specifying the number of levels of children to display. The line of output for each child will have the sum of that child and its children to the depth specified for the ADD option. 1 is the default value. Therefore, if *n* is omitted, direct children each have a line on the report. If *n* is 2, direct children and grandchildren each have a line on the report output. ALL is a synonym for 99.

`ADD`

Sums the hierarchy to the depth specified by *m* for each line generated by the GET or WITH CHILDREN command.

m|ALL

Is a positive integer from 1 to 99 specifying the number of levels of children to consolidate on each line of the report output. If a number greater than 99 is specified, a warning message is displayed and *m* is set to 99. ALL is the default value. Therefore, if *m* is omitted, the consolidated line will sum all children. If *m* is 2, direct children and grandchildren will be consolidated for each line on the report output. ADD 99 aggregates children to 99 levels. ALL is a synonym for 99.

`CAPTION`

Indicates that the caption of the parent value will appear for the total row.

Note: The AS CAPTION phrase is supported for tagged rows, including those that do not use the GET/WITH CHILDREN or ADD syntax. However, the hierarchy must be defined (by specifying the PARENT_OF attribute) in order to load and display the caption values. If the hierarchy is not defined, the AS CAPTION phrase is ignored.

'text'

Is a text string to use as the row title for the aggregate row. The CAPTION field defined in the Master File will not be used as the caption on the report output.

label

Is an explicit row label. Each generated row is labeled with the specified label text.

Example: Displaying Consolidated Data For Multiple Hierarchy Levels

In the following request, the first WITH CHILD command displays the detail data for the hierarchy starting with account 3100. The next WITH CHILD command creates a consolidated line for the parent account (3100) and each direct child:

```
SET BLANKINDENT = ON
SET FORMULTIPLE = ON
JOIN SYS_ACCOUNT IN CENTGL TO ALL SYS_ACCOUNT IN CENTSYSF
TABLE FILE CENTGL
 SUM NAT_AMOUNT/D10.0 NAT_YTDAMT/D10.0
 FOR GL_ACCOUNT
 3100 WITH CHILDREN ALL AS CAPTION      OVER
 " "                                    OVER
 BAR AS =                               OVER
 " "                                    OVER
 3100 WITH CHILDREN ADD AS CAPTION
 IF PERIOD EQ '2002/03'
END
```

In the following output, the top portion shows the detail level data. The bottom portion shows the consolidated data. In the consolidated portion of the report:

- There is one line for the parent that is the sum of itself plus all of its children to all levels.
- There is one line for each direct child of account 3100 (Selling Expenses): Advertising, Promotional Expenses, Joint Marketing, and Bonuses/Commissions.

- The line for Advertising is the sum of itself plus all of its children. If it had multiple levels of children, they would all have been added into the sum. The other direct children of 3100 did not themselves have children, so the sum on each of those lines consists of only the parent value.

	Month Actual	YTD Actual
Selling Expenses	.	.
Advertising	.	.
TV/Radio	1,049,146.	2,954,342.
Print Media	244,589.	721,448.
Internet Advertising	9,542.	29,578.
Promotional Expenses	53,719.	151,732.
Joint Marketing	97,135.	289,799.
Bonuses/Commisions	100,188.	304,199.
Selling Expenses	1,554,319.	4,451,098.
Advertising	1,303,277.	3,705,368.
Promotional Expenses	53,719.	151,732.
Joint Marketing	97,135.	289,799.
Bonuses/Commisions	100,188.	304,199.

Syntax: How to Display the Same Record in Multiple FOR Rows

```
SET FORMULTIPLE = {ON|OFF}

ON TABLE SET FORMULTIPLE = {ON|OFF}
```

where:

ON

Enables you to reference the same value of a FOR field in more than one row in an FML request.

With FORMULTIPLE set to ON, a value retrieved from the data source is included on every row in the report output for which it matches the tag references.

OFF

Does not allow you to include the same value in multiple rows. If there are multiple FOR rows present in an FML request for the same record, and FORMULTIPLE is OFF, the record will only be included in one of the FOR rows. OFF is the default value.

With FORMULTIPLE set to OFF, multiple tags referenced in any of these ways (OR, TO, *) are evaluated first for an exact reference or for the end points of a range, next for a mask, and finally within a range. For example, if a value is specified as an exact reference and then as part of a range, the exact reference is displayed. Note that the result will be unpredictable if a value fits into more than one row whose tags have the same priority (for example, an exact reference and the end point of a range.)

Indenting Row Titles in an FML Hierarchy

To clarify relationships within an FML hierarchy, the captions (titles) of values are indented at each level. You can use the SET BLANKINDENT command in an HTML, PDF, or PostScript report to specify the indentation between each level in the hierarchy. You can use the default indentation for each hierarchy level or choose your own indentation value. To print indented captions in an HTML report, you must set the BLANKINDENT parameter to ON or to a number.

In Release 5.2.3, you can also change the number of blank spaces before the parent line of a hierarchy or before any FML tag or RECAP row. For more information, see *Indent an FML Label, Tag, or Caption* on page 1-91.

Syntax: How to Indent Row Titles in an FML Hierarchy

```
SET BLANKINDENT = {ON|OFF|n}

ON TABLE SET BLANKINDENT {ON|OFF|n}
```

where:

`ON`

Indents FML hierarchy captions 0.125 units for each space that would normally appear before the caption. For child levels in an FML hierarchy, it indents 0.125 units for each space that would normally display between this line and the line above it.

`OFF`

Turns off indentations for FML hierarchy captions in an HTML report. For other formats, uses the default indentation of two spaces. OFF is the default value.

n

Is an explicit measurement in the unit of measurement defined by the UNITS parameter. This measurement is multiplied by the number of spaces that would normally appear before the caption. For child levels in an FML hierarchy, *n* units are indented for each space that would normally appear between this line and the line above it. 2 is the default value. Zero (0) produces the same report output as OFF. Negative values for *n* are not supported. They generate the following message, and the request processes as if BLANKINDENT = OFF:

```
VALID VALUES ARE OFF, ON OR A POSITIVE NUMBER (IN CURRENT UNITS)
```

You can change the number of blank spaces before the parent line of a hierarchy or before any FML tag or RECAP line in an FML request.

For an illustration of the default indent, see *Displaying an FML Hierarchy With Captions* on page 1-77.

Returning FOR Field Values in a Computed Column

How to:

Retain FOR Field Values for Consolidated Report Rows

Example:

Reversing Negative Signs of FOR Field Values in an FML Report

In an FML report that contains consolidated data, you can use the FMLINFO function in a COMPUTE command to accurately determine the value of the FOR field for each row. You can use this function to:

- Change the signs of financial data such as revenue figures, which are often stored as negative (-) values, into positive values for presentation on a report.
- Pass the FOR field value as a parameter in a drill down to another report. If row captions are displayed in the report, the drill down can be implemented from the caption.

Note: The SET FORMULTPLE parameter must be set to ON in order to use the FMLINFO function. This enables an incoming record to be used in more than one row of an FML report.

Syntax: How to Retain FOR Field Values for Consolidated Report Rows

```
FMLINFO('FORVALUE', outfield)
```

where:

`'FORVALUE'`

Alphanumeric

Returns the FOR value associated with each row in an FML report. If the FML row was generated as a sum of data records using the OR phrase, FMLINFO returns the first FOR value specified in the list of values. If the OR phrase was generated by an FML hierarchy ADD command, FMLINFO returns the FOR value associated with the parent specified in the ADD command.

`outfield`

Alphanumeric

Is the name of the field that contains the result, or the format of the output value enclosed in single quotation marks.

Example: Reversing Negative Signs of FOR Field Values in an FML Report

The following request creates a field called PRINT_AMT that reverses the negative signs in the NAT_AMOUNT field for account numbers less than 2500 in the CENTSYSF data source. The CENTGL data source contains the hierarchy information for CENTSYSF. Therefore, CENTGL is joined to CENTSYSF for the request:

```
SET FORMULTIPLE = ON
JOIN SYS_ACCOUNT IN CENTGL TO ALL SYS_ACCOUNT IN CENTSYSF
TABLE FILE CENTGL
 SUM NAT_AMOUNT/D10 IN 30
 COMPUTE PRINT_AMT/D10 = IF FMLINFO('FORVALUE', 'A7') LT '2500'
  THEN 0-NAT_AMOUNT ELSE NAT_AMOUNT;
 COMPUTE FORV/A4 = FMLINFO('FORVALUE', 'A4');
 COMPUTE ACTION/A9 = IF FORV LT '2500'  THEN 'CHANGED' ELSE 'UNCHANGED';
 FOR GL_ACCOUNT
 2000 WITH CHILDREN 2 ADD AS CAPTION
END
```

Note that the parent value specified in the WITH CHILDREN ADD command (2000) is returned for the first row on the report. Each subsequent row is also a consolidated subsection of the hierarchy with a parent value that is returned by FMLINFO:

	Month Actual	PRINT_AMT	FORV	ACTION
Gross Margin	-25,639,223	25,639,223	2000	CHANGED
Sales Revenue	-62,362,490	62,362,490	2100	CHANGED
Retail Sales	-49,355,184	49,355,184	2200	CHANGED
Mail Order Sales	-6,899,416	6,899,416	2300	CHANGED
Internet Sales	-6,107,890	6,107,890	2400	CHANGED
Cost Of Goods Sold	36,723,267	36,723,267	2500	UNCHANGED
Variable Material Costs	27,438,625	27,438,625	2600	UNCHANGED
Direct Labor	6,176,900	6,176,900	2700	UNCHANGED
Fixed Costs	3,107,742	3,107,742	2800	UNCHANGED

Formatting an FML Report

How to:

Format a Row, Column, or Cell in an FML Report

Indent an FML Label, Tag, or Caption

Example:

Formatting a Cell in an FML Matrix

Indenting FML RECAP Rows

You can format a row or cell, as well as a column, in an FML report.

In Release 5.2.3, you can indent TAG rows, RECAP rows, or all rows in a dynamically generated hierarchy.

Syntax: How to Format a Row, Column, or Cell in an FML Report

```
TYPE = type, [COLUMN = column] [LABEL = {Rn|label}], format_def ,$
```

where:

type

Identifies the component you wish to format. Possible values are:

REPORT denotes a row with the specified label.

DATA denotes a row with the specified label, which contains user-supplied data values.

FREETEXT denotes a free text or blank row with the specified label.

UNDERLINE denotes underlines generated by BAR. Formatting of an underline is supported for PDF and PS, but not for HTML reports.

column

Identifies a specific column. You can identify the column by its name or its position in a row.

LABEL

Is the controlling factor in identifying and formatting an FML row.

Note that the label is used to identify a row for calculation or formatting. The label for a TAG or DATA row will never appear in the report output; it is used only to identify rows within the FML code. For a RECAP row, the name of the calculated value serves as a label; it will appear in the report unless an alternate title is specified. The label can be implicit or explicit.

Rn

Is an implicit row label. It is assigned automatically by FML, and retained internally for processing even if you set an explicit label. To determine the value of *n*, count the number of rows up to and including the desired row.

label

Is an explicit row label that you can assign to identify a row more clearly.

format_def

Is the formatting definition, such as FONT, SIZE, STYLE, and COLOR.

Note: To format a cell, identify the cell as the intersection of a column and row using COLUMN =… plus LABEL = … in the same StyleSheet declaration.

Example: Formatting a Cell in an FML Matrix

The following request generates a report in which the data value for AMOUNT is bold in the row titled CASH; however, the row title CASH is not bold. This is accomplished by pinpointing the cell in the StyleSheet declaration—that is, the column (N2) within the row (CA).

```
SET PAGE-NUM = OFF
TABLE FILE LEDGER
 SUM AMOUNT FOR ACCOUNT
 10$$ AS 'CASH'                 LABEL CA     OVER
 1100 AS 'ACCOUNTS RECEIVABLE' LABEL AR     OVER
 1200 AS 'INVENTORY'            LABEL INV    OVER
 RECAP CURASST/I5C = CA + AR + INV;
 ON TABLE SET STYLE *
 TYPE = REPORT, GRID = OFF ,$
 TYPE = REPORT, COLUMN = N2, LABEL = CA, STYLE = BOLD ,$
 ENDSTYLE
END
```

The output is:

	AMOUNT
CASH	**21,239**
ACCOUNTS RECEIVABLE	18,829
INVENTORY	27,307
CURASST	67,375

For additional examples of row- and cell-based formatting, see *Formatting an FML Report* in the *Creating Financial Reports* chapter in the *Creating Reports With WebFOCUS Language* manual.

Syntax: How to Indent an FML Label, Tag, or Caption

```
FOR forfield [IN k]
tag [[GET CHILDREN|WITH CHILDREN] n] INDENT m [AS ['text'|CAPTION]] [OVER]
```

or

```
RECAP fieldname[/format] = expression; INDENT m [AS 'text']
```

where:

forfield

Is a field in the data source whose values will be included in the report.

k

Is the starting column for the FOR value in an FML report.

tag

Is a value of *forfield* to be displayed on a row of the FML report.

n

Is the number of levels of an FML hierarchy to display on the FML report.

m

Is a positive integer (zero is not supported) specifying the number of spaces to indent the tag value, label, or caption of an FML row or hierarchy. The indentation starts from column one if there is no IN phrase specified in the FOR command. If there is an IN phrase on the FOR command, indentation starts from the column specified in the IN phrase. The maximum indentation is the same as the maximum length of an AS name.

If you indent an FML hierarchy, the parent line of the hierarchy is indented the number of spaces specified as the indent. The hierarchy levels are indented two spaces from each other. If the GET CHILDREN phrase is used, the first line of the hierarchy is indented an additional two spaces because the hierarchy output begins with the first child rather than the parent.

text

Is a label to be displayed on a row of the FML report.

CAPTION

Indicates that a caption field has been defined in the Master File.

OVER

Indicates that this row is not the last row to be displayed.

fieldname

Is a name you assign to the value calculated by the RECAP command.

format

Is the RECAP field's USAGE format. It cannot exceed the column width. The format of the column in which the calculated value will be displayed is the default value.

expression

Is the expression that describes how to calculate the RECAP field's value.

Example: Indenting FML RECAP Rows

The following request sums price, cost, and quantity in stock for digital and analog product types. The first RECAP command calculates the total for each column and indents the label five spaces. The second RECAP command calculates the profit and indents the label 10 spaces:

```
SET BLANKINDENT = ON
SET FORMULTIPLE = ON
TABLE FILE CENTINV
 SUM PRICE COST QTY_IN_STOCK
 FOR PRODTYPE
 Digital                                                OVER
 Analog                                                 OVER
 BAR                                                    OVER
 RECAP TOTAL = R1 + R2; INDENT 5 AS 'Total:'            OVER
 BAR                                                    OVER
 RECAP PROFIT(2) = TOTAL(1) - TOTAL(2); AS 'Profit:' INDENT 10
END
```

The output is:

	Price:	Our Cost:	Quantity In Stock:
Digital	4,080.00	3,052.00	119143
Analog	1,883.00	1,371.00	139345
Total:	5,963.00	4,423.00	258488
Profit:		1,540.00	

Graph Enhancements

The graph environment has been enhanced to:

- Facilitate saving a graph as a GIF file in a server environment.
- Alter X- and Y-axis values in a SCATTER graph using linear regression.
- Offer a variety of new graph type options.
- Send graph output directly to a printer.

Graphing capabilities are fully documented in the *Creating a Graph* chapter in the *Creating Reports With WebFOCUS Language* manual.

For a graphical implementation, see the *Creating a Graph With Graph Assistant* chapter in the *Creating Reports With Graphical Tools* manual.

Saving a Graph as a GIF File

How to:

Save a Graph as a GIF File

Example:

Inserting a GIF Image Into a PDF Report

The SET GRAPHSERVURL command eliminates the need to set the IBIJAVAPATH system environment variable before holding or saving a graph image. This enables users who are running against a server environment where the WebFOCUS Reporting Server is installed on an OS/390 and z/OS, Windows NT/2000/XP, or UNIX machine to save graph output as a GIF file. GIF images can be embedded in a PDF or HTML report.

The GRAPHSERVURL parameter sends an http request to the machine that has the WebFOCUS Graph Servlet. The graph image is created by the WebFOCUS Graph Servlet, and the image is sent back to a temporary location on the WebFOCUS Reporting Server (if a FILEDEF has not been specified), or to the location specified in a FILEDEF.

Syntax: How to Save a Graph as a GIF File

```
[FILEDEF filename DISK drive:\…\filename.GIF]
SET GRAPHSERVURL = graph_servlet_URL
GRAPH FILE file
 graph commands
 ON GRAPH HOLD AS filename FORMAT GIF
END
```

where:

`FILEDEF`

Saves the GIF file to the location you specify.

filename

Is the name you give the GIF file. If you want to prompt for a file name, include an amper variable such as &FILENAME in the procedure.

Note: To insert a GIF image that resides in a permanent location, you must provide the fully qualified path to the GIF file. For example, in the following code

```
TYPE = REPORT, IMAGE = drive:\…\filename.gif
```

drive:\…\filename.gif is the path where the GIF file is located. The WebFOCUS Reporting Server must be installed on that drive.

graph_servlet_URL

Is the URL to invoke the WebFOCUS Graph Servlet.

file

Is the name of the data source you wish to report against.

Example: Inserting a GIF Image Into a PDF Report

The following illustrates how to create a GIF file in a graph request, then embed the GIF image into a PDF report.

1. Create the following remote procedure in a location accessible to the application path.

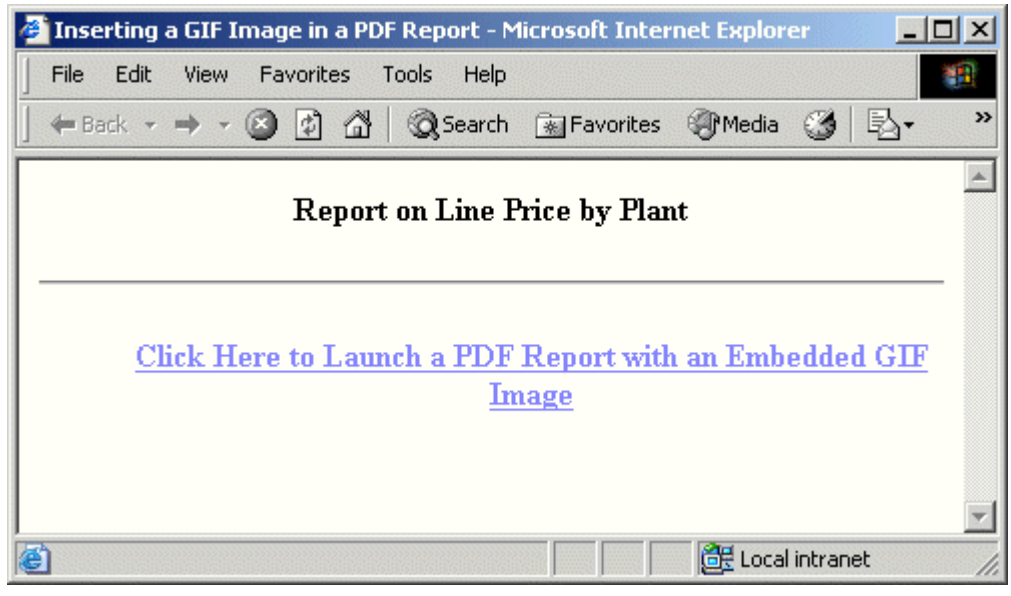
```
SET GRAPHSERVURL = http://localhost/ibi_apps/IBIGraphServlet
GRAPH FILE CENTORD
 SUM LINEPRICE
 ACROSS PLANTLNG AS 'Plant'
 ON GRAPH HOLD AS PLANT FORMAT GIF
END
TABLE FILE CENTORD
 SUM LINEPRICE
 BY PLANTLNG AS 'Plant'
 ON TABLE SET STYLE *
 TYPE = REPORT, IMAGE = plant.gif, POSITION = (4 0), SIZE = (5 3) ,$
 ENDSTYLE
 ON TABLE PCHOLD FORMAT PDF
END
```

2. Save the procedure as HOLDGIF.

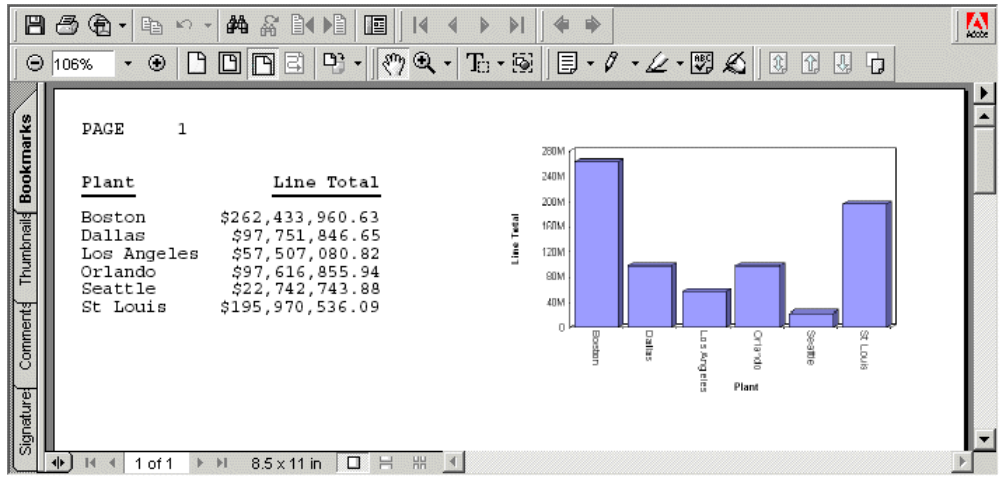
3. Create the following HTML file, which calls the HOLDGIF WebFOCUS procedure from a hyperlink:

```
<HTML>
<HEAD>
<TITLE> Inserting a GIF Image in a PDF Report </TITLE>
</HEAD>
<BODY>
<H4 ALIGN=CENTER>Report on Line Price by Plant</H4>
<HR>
<P><FONT SIZE=+2></FONT></P>
<UL TYPE=SQUARE
<LI><A HREF="http://localhost/ibi_apps/WFServlet?IBIF_ex=holdgif">
<H4 ALIGN=CENTER>Click Here to Launch a PDF Report with an Embedded
    GIF Image</H4></A>
</UL>
</BODY>
</HTML>
```

The resulting launch page looks like this:


Inserting a GIF Image in a PDF Report - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Report on Line Price by Plant
Click Here to Launch a PDF Report with an Embedded GIF Image
Local intranet

4. Click the link on the launch page to run the report. The output looks like this:



You can run the report from a browser or from Developer Studio. To distribute the report using ReportCaster, you would schedule the actual procedure (in this example, HOLDGIF).

Altering X- and Y-Axis Values Using Linear Regression

You can use basic linear regression to alter the X- and Y-axis values in a SCATTER graph with the SET GTREND command. Basic linear regression involves the average of the summation of X- and Y-axis values to determine a linear equation that expresses the trend of the scatter diagram.

Syntax: How to Alter X- and Y-Axis Values Using Linear Regression

To alter the X- and Y-axis values in a SCATTER graph, the syntax is

```
SET GTREND = {ON|OFF}
```

where:

ON

Uses basic linear regression to alter the X- and Y-axis values.

OFF

Does not alter the X- and Y-axis values. OFF is the default value.

New Graph Type Options

The following graph styles are supported when using the SET LOOKGRAPH command in a GRAPH request:

- **GANTT.** Provides a visual representation of project-oriented time-critical events. Gantt charts require six display fields and one sort field, in that order. Conditional styling and drill down are not supported for GANTT charts.
- **POSITION.** Product position charts provide a visual representation of market share and growth versus revenue and measurement (past, present, future). Product position charts require a set of three display fields. A sort field is not required.
- **VWATERFL.** There are no field column limitations for vertical waterfall charts.
- **HWATERFL.** There are no field column limitations for horizontal waterfall charts.
- **PARETO.** Displays data following Pareto's 80:20 rule. Pareto charts require only one display field.
- **Multiple Y-axis charts.** Stacks charts in order to make them easier to read, analyze, and manage. You can have a MULTI3Y, MULTI4Y, or MULTI5Y chart, each with the corresponding number of Y-axis fields.

Sending Graph Output Directly to a Printer

You can send graph output directly to a printer. Add the following syntax to your GRAPH request:

```
ON GRAPH SET PRINT OFFLINE
```

Expanded Limits

In this section:

Increased Number of Display Fields
Increased Number of Columns Generated by ACROSS
Increased Numbers of Headings and Footings
Increased Space for Column Titles

You can include an increased number or length of certain elements in a report. In Release 5.2.3, you can also create a report with an unlimited number of -INCLUDE commands.

Increased Number of Display Fields

The number of display fields you can include in a report has increased from 495 to approximately 1024 (495 for MATCH requests). This does not include sort fields. The maximum number of fields in a request is affected by the following factors:

- Hidden (NOPRINT) fields.
- Temporary fields (created with the COMPUTE and DEFINE commands).
- Internal fields. For example, TABPAGENO.
- The size of the field.
- Field references in headings and footings.

Increased Number of Columns Generated by ACROSS

In Release 5.2.3, each ACROSS phrase can generate up to 1,056 columns of data. The total number of ACROSS columns is equal to the total number of ACROSS sort field values multiplied by the total number of display fields.

Increased Numbers of Headings and Footings

In Release 5.2.3, the following guidelines apply to report headings and footings, page headings and footings, and sort headings and footings:

- In a single report, there can be a maximum of 32K characters for all types of heading and footing text.
- The maximum number of sort headings plus sort footings in one request is 64.
- The maximum number of nested headings is 64.

Increased Space for Column Titles

In Release 5.2.3, you can enter column title text of unlimited length.

Prompting for Variables in CGI/Servlet

How to:

Add a Description of the Procedure

Add a Heading to the Form

Add a Variable Description to the Form

Set a Default Variable Value

Add a Static Single-Select List of Values

Add a Dynamic Single-Select List of Values

Add a Static Multi-Select List of Values

Add a Dynamic Multi-Select List of Values

Add a Range of Values List

Example:

Adding a Single-Selectable List of Values

The amper auto prompting feature enables you to prompt users for the variables necessary to execute a procedure. With this feature, you can also design the form that will prompt for the amper variables.

You can add the following with auto prompting:

- Description of the procedure. For details, see *Add a Description of the Procedure* on page 1-100.
- Heading for the form. For details, see *Add a Heading to the Form* on page 1-100.
- Description of a variable. For details, see *Add a Variable Description to the Form* on page 1-100.
- Default variable values. For details, see *Set a Default Variable Value* on page 1-100.
- Static or dynamic single-select list. For details, see *Add a Static Single-Select List of Values* on page 1-101 and *Add a Dynamic Single-Select List of Values* on page 1-101.
- Static or dynamic multi-select list. For details, see *Add a Static Multi-Select List of Values* on page 1-101 and *Add a Dynamic Multi-Select List of Values* on page 1-102.
- Range of values.

Auto prompting is not supported for Managed Reporting.

Syntax: How to Add a Description of the Procedure

```
-<description>text</description>
```

where:

text

Is the description of the procedure.

Note: If the text takes up more than one line, you must repeat the <description> and </description> tags.

Syntax: How to Add a Heading to the Form

```
-<heading>text</heading>
```

where:

text

Is the heading that will display on the form.

Syntax: How to Add a Variable Description to the Form

```
'&variable.description.'
```

where:

&variable

Is the variable name, including the ampersand, whose value is being prompted for.

description

Is a description of the variable that replaces the variable name in the prompt.

Syntax: How to Set a Default Variable Value

```
-DEFAULT &variable = value
```

where:

&variable

Is the variable, including the ampersand, for which you are setting a default value.

value

Is the default value for the variable.

Syntax: How to Add a Static Single-Select List of Values

```
'&variable.(value, value2[,value3][,value4]...).[description.]'
```

where:

&variable

Is the variable, including the ampersand, for which you are supplying a list of values.

value, value2, value3, value4...

Are the values comprising the list of selectable variable values.

description

Is an optional description of the variable.

Syntax: How to Add a Dynamic Single-Select List of Values

```
'&variable.(FIND fieldname IN datasource).[description.]'
```

where:

&variable

Is the variable, including the ampersand, for which you are supplying a list of values.

fieldname

Is the field name containing the possible variable values.

datasource

Is the data source that contains the field name specified in *fieldname*.

description

Is an optional description of the variable.

Syntax: How to Add a Static Multi-Select List of Values

```
'&variable.({AND|OR} (value1,value2,...valuen)).[description.]'
```

where:

&variable

Is the variable, including the ampersand, for which you are supplying a list of values.

value1, value2, value3, value4...

Are the values comprising the list of selectable variable values.

description

Is an optional description of the variable.

Syntax: How to Add a Dynamic Multi-Select List of Values

```
'&variable.({AND|OR} (FIND fieldname IN datasource)).[description.]'
```

where:

&variable

Is the variable, including the ampersand, for which you are supplying a list of values.

fieldname

Is the field name containing the possible variable values.

datasource

Is the data source that contains the field name specified in *fieldname*.

description

Is an optional description of the variable.

Syntax: How to Add a Range of Values List

```
'&variable.(FROM range1 TO range2)'
```

where:

&variable

Is the numerical variable, including the ampersand, for which you are supplying a list of values.

range1

Is the starting numerical value of the range of list of values.

range2

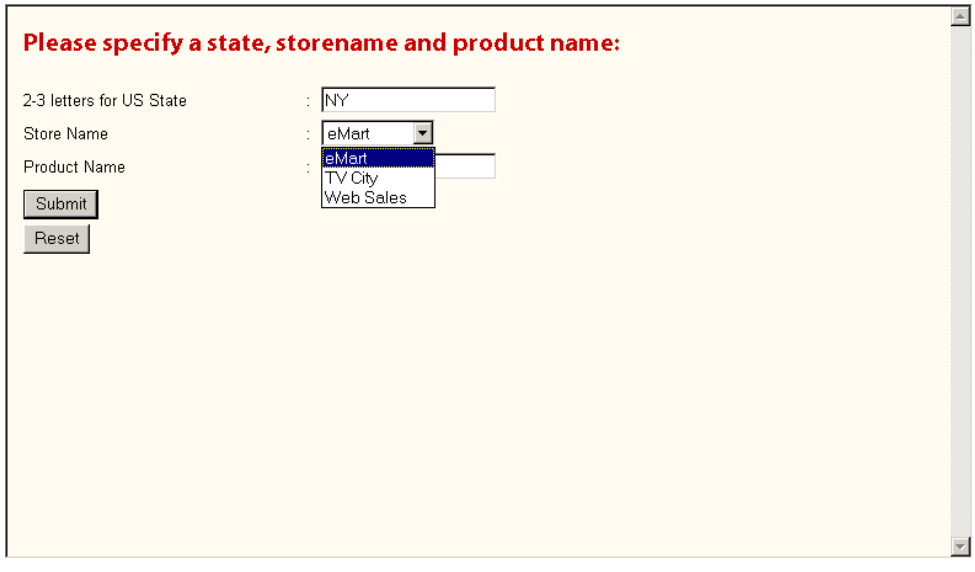
Is the ending numerical value of the range of list of values.

Example: Adding a Single-Selectable List of Values

The following request generates a form that lists valid values for the SNAME field. The user can select one value from the list. The State, NY, is supplied by default. Product name is supplied by the user.

```
-<heading> Please specify a state, storename and product name: </heading>
-DEFAULT &STATE=NY
TABLE FILE CENTORD
 SUM QTY_IN_STOCK BY STATE BY PRODNAME BY SNAME
 ON TABLE SUBHEAD
 "Inventory Report"
 WHERE STATE EQ '&STATE.2-3 letters for US State.'
 WHERE SNAME EQ '&SNAME.(eMart,TV City,Web Sales).Store Name.'
 WHERE PRODNAME EQ '&PRODNAME.Product Name.'
END
```

The HTML form generated by this code is:



For examples of a description of a procedure, static and dynamic multi-select lists, and range of values, see *Amper Auto-Prompting in CGI/Servlet* in the *Coding a User Interface* chapter in the *Developing Reporting Applications* manual.

Including Script Files in Reports

The SET JSURL command enables you to include JavaScript or VBScript files in an HTML report.

Syntax: How to Include a JavaScript or VB Script File in an HTML Report

```
SET JSURL = '/file1 [/file2] [/file3]...'
```

where:

/file1 [/file2] [/file3]...

Are the files that contain JavaScript or VBScript. If there is more than one JavaScript file, the delimiter is a blank and the values must be enclosed in single quotation marks. Files must be in a location that is accessible by the Web server.

You can reference files with a URL.

Specifying Holidays

You can use the SET HDAY command to specify company-specific holidays in a file for later retrieval. The dates that are designated as holidays can be used with the date functions DATEDIF, DATEMOV, DATECVT, and DATEADD. The file must be named HDAY, followed by four characters.

Syntax: How to Specify Holidays

```
SET HDAY = xxxx
```

where:

xxxx

Are the letters in the name of the holiday file, named HDAY*xxxx*. This string must be four characters long.

By default, this parameter is not set.

Establishing a Non-Standard Alias

In Release 5.2.3, you can generate an alias other than /ibi_html on the WebFOCUS Reporting Server using the SET FOCHTMLURL command.

This command is usually included in a profile or in one of the WFS files (site.wfs) to establish a default setting for the installation.

Syntax: How to Access a Resource With a Non-Standard Alias

```
SET FOCHTMLURL = /my_html
```

where:

my_html

Is an alias other than /ibi_html that is generated on the WebFOCUS Reporting Server.

Compiling Expressions Containing DEFINEs

How to:

Compile DEFINE Expressions

In Release 5.2.3, you can use the DEFINE compiler to compile expressions into machine code for faster processing. Benefits of the DEFINE compiler include:

- Compilation of only those expressions that are actually used in the report request.
- Much faster execution of expressions containing complex calculations in long packed fields.
- Compilation of date expressions.

Once the DEFINE compiler is invoked, any request that uses a DEFINE expression causes the expression to be compiled and then loaded into the system. For each record of the request that needs the computation, the system executes the generated code. This compiler is most effective with report requests that include a large number of virtual (DEFINE) fields and read a large number of records, because the speed of evaluation per record in such requests offsets the extra compilation and load steps.

Syntax: How to Compile DEFINE Expressions

Issue the following command in a profile or procedure

```
SET DEFINES = {COMPILED|OLD}
```

where:

`COMPILED`

Implements expression compilation at request run time, compiling only those virtual fields (DEFINEs) that are used in the request. COMPILED is the default value in the OS/390 and z/OS environments.

`OLD`

In the UNIX and Windows NT environments, this indicates that expression compilation will not take place. In the Mainframe environment, the OLD setting leaves expression compilation up to the control of the current value of the SET COMPUTE command. OLD is the default in the UNIX, Windows NT, OS/390 and z/OS, and VM environments.

Note: Linux running in any operating environment uses the UNIX version of the compiler and has the same default value.

The SET DEFINES command is not supported in an ON TABLE phrase.

Exiting on Error

Using the SET ERROROUT command, you can control how a server responds to error conditions encountered in a procedure.

Syntax: How to Control Error Processing

Enter the following in a profile or procedure

```
SET ERROROUT = {ON|OFF}
```

where:

`ON`

Terminates the procedure and returns a message to the client in the event of an error on the server.

`OFF`

Continues processing when an error is encountered. OFF is the default value.

Converting Case for File Names and Directories

Both UNIX and WebFOCUS support uppercase, lowercase, and mixed-case file names and directories. Since lowercase is the default in UNIX environments, WebFOCUS offers a way to automatically convert file names and directories to lowercase.

Syntax: How to Convert File Names to Lowercase

```
SET FILECASE = {ACTUAL|LOWER}
```

where:

ACTUAL

Retains file names in uppercase, lowercase, or mixed-case, exactly as they are entered. When this setting is used, WebFOCUS looks for file names exactly as they are entered (uppercase, lowercase, or mixed-case).

LOWER

Converts uppercase or mixed-case names to lowercase. When this setting is used, WebFOCUS looks for file names in lowercase. LOWER is the default value.

Adding Borders for Emphasis

How to:

Add and Format Borders

Example:

Inserting and Formatting a Border

You can add a border around an entire HTML, PDF, or PS report, or just add a border around headings, footings, and columns to emphasize them. Using the BORDER attribute in a StyleSheet, you can also specify the weight, style, and color of border lines. If you wish, you can specify formatting variations for the top, bottom, left, and right borders.

For a graphical implementation using the Report Painter in Developer Studio, see *Adding Borders to an HTML or PDF Report* in the *Styling Reports With Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.

Syntax: How to Add and Format Borders

To request a uniform border, use this syntax

```
TYPE = type, BORDER = option, [BORDER-STYLE = line_style,]
 [BORDER-COLOR = {color|RGB(r g b)},] $
```

To specify different characteristics for the top, bottom, left, and/or right borders, use this syntax

```
TYPE = type, BORDER-position = option,
 [BORDER[-position]-STYLE = line_style,]
 [BORDER[-position]-COLOR = {color|RGB(r g b)},] $
```

where:

type

Identifies the report component to which borders are applied.

option

Can be one of the following values:

ON turns borders on. ON generates the same line as MEDIUM.

Note: The MEDIUM line setting ensures consistency with lines created with GRID attributes.

OFF turns borders off. OFF is the default value.

LIGHT specifies a thin line.

MEDIUM identifies a medium line. ON sets the line to MEDIUM.

HEAVY identifies a thick line.

width specifies the line width in points, where 72 pts=1 inch.

Tip: Line width specified in points is displayed differently in HTML and PDF output. For uniform appearance, regardless of display format, use LIGHT, MEDIUM, or HEAVY.

line_style

Sets the style of the border line. WebFOCUS StyleSheets support all of the standard Cascading Style Sheet line styles. Several 3-dimensional styles are available only in HTML, as noted by asterisks. Possible values are:

Style	Description
NONE	No border is drawn.
SOLID	Solid line.
DOTTED	Dotted line.
DASHED	Dashed line.

Style	Description
DOUBLE	Double line.
GROOVE*	3D groove.
RIDGE*	3D ridge.
INSET*	3D inset.
OUTSET*	3D outset.

color

Is one of the preset color values. BLACK is the default value.

If the display or output device does not support colors, it substitutes shades of gray. For a complete list of available color values, see *Color Values in a Report* in the *Formatting Report Data* chapter in the *Creating Reports With WebFOCUS Language* manual.

RGB

Specifies the font color using a mixture of red, green, and blue.

(r g b)

Is the desired intensity of red, green, and blue, respectively. The values are on a scale of 0 to 255, where 0 is the least intense and 255 is the most intense. Using the three color components in equal intensities results in shades of gray.

position

Specifies which border line to format. Possible values are: TOP, BOTTOM, LEFT, RIGHT.

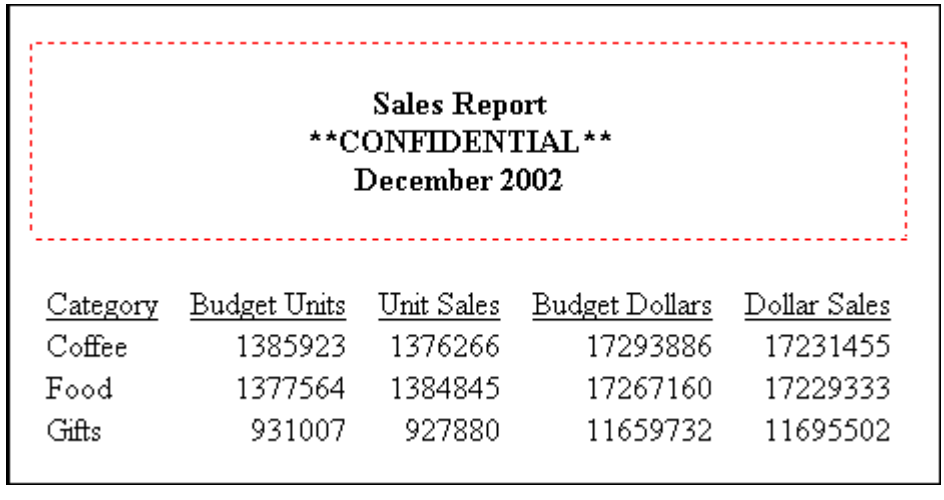
You can specify a position qualifier for any of the BORDER attributes. This enables you to format line width, line style, and line color individually, for any side of the border.

Example: Inserting and Formatting a Border

The following request generates an HTML report with a light red dotted line around the entire report heading.

```
TABLE FILE GGSALES
 SUM BUDUNITS UNITS BUDDOLLARS DOLLARS
 BY CATEGORY
 ON TABLE SUBHEAD
 "</1 Sales Report"
 "**CONFIDENTIAL**"
 "December 2002 </1"
 ON TABLE SET PAGE-NUM OFF
 ON TABLE SET ONLINE-FMT HTML
 ON TABLE SET HTMLCSS ON
 ON TABLE SET STYLE *
 TYPE = REPORT, GRID = OFF ,$
 TYPE = TABHEADING, STYLE = BOLD, JUSTIFY = CENTER, BORDER = LIGHT,
  BORDER-COLOR = RED, BORDER-STYLE = DASHED ,$
 ENDSTYLE
END
```

The output is:



Sales Report
****CONFIDENTIAL****
December 2002

Category	Budget Units	Unit Sales	Budget Dollars	Dollar Sales
Coffee	1385923	1376266	17293886	17231455
Food	1377564	1384845	17267160	17229333
Gifts	931007	927880	11659732	11695502

Tip: You can use the same BORDER syntax to generate this output in a PDF or PS report.

Aligning Decimals in a Multi-Line Heading

How to:

Specify Decimal Alignment of a Heading or Footing

Measure for Column Width and Decimal Alignment

Example:

Aligning Data and Text in a Multi-Line Heading or Footing

You can align decimal points in a multi-line heading or footing. This feature is particularly useful when you are working with currencies that have different display conventions. For example, if a figure is in dollars, it is formatted with a decimal point and two places for zeros. If it is in Swiss Francs, it is formatted with a decimal place and four zeros. If it is in Yen, it is formatted with the decimal at the end and no zeros.

By aligning the decimal points in a vertical stack, you can more easily read and compare these numbers, as illustrated in the following output:

Floating decimal points		**Aligned decimal points**	
Bond	Face Value	Bond	Face Value
Galosh Ltd.	22375.5784596	Galosh Ltd.	22375.5784596
Mukluk Inc.	1212345.457	Mukluk Inc.	1212345.457
Overshoe Inc.	232.45484	Overshoe Inc.	232.45484

This technique uses a width specification for the item that contains decimals, combined with a variation on standard left/right/center justification to achieve the proper decimal alignment.

Decimal alignment is supported for styled reports in the following formats: HTML (requires Cascading Style Sheets: SET HTMLCSS ON), PDF, and PostScript.

For the graphical implementation of this feature, see *Adding a Page Heading or Footing* in the *Creating Reports With Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.

For related information about aligning text and data in headings and footings, see *Aligning Content in a Multi-Line Heading or Footing* in the *Using Heading, Footing, Titles, and Labels* chapter in the *Creating Reports With WebFOCUS Language* manual.

Syntax: How to Specify Decimal Alignment of a Heading or Footing

To align the decimal points, define the width of the decimal item, then measure how far in from the right side of a column you want to position the decimal point. This places the decimal point in the same position in a column, regardless of the number of decimal places displayed to its right.

```
TYPE = hcomponent, ITEM = n, WIDTH = w, JUSTIFY = DECIMAL(m) ,$
```

where:

hcomponent

Specifies the type of heading component that should display with decimal alignment. Possible values are:

`TABHEADING` specifies the report heading.

`TABFOOTING` specifies the report footing.

`HEADING` specifies the page heading.

`FOOTING` specifies the page footing.

`SUBHEAD` specifies a sort heading.

`SUBFOOT` specifies a sort footing.

n

Specifies the part of the heading or footing to align by its position within the heading or footing. Note that several other heading and footing subcomponents can also be used to select a specific item.

w

Specifies the width of the item to be aligned in the units specified by the UNITS parameter (inches by default).

m

Is the measurement in the unit of measurement specified by the UNITS parameter (inches by default), which specifies how far in from the right side of a column you want to place your decimal point. With this specification, you can locate the decimal point in the same position within a column, regardless of the number of decimal places displayed to its right. Text can also be aligned using this technique. The rightmost character aligns in the position immediately to the left of the decimal point.

The measurement must be a portion of the width you have specified for this item.

If you wish to mix fonts while retaining the decimal alignment, see *Measure for Column Width and Decimal Alignment* on page 1-115 for additional instructions.

Example: Aligning Data and Text in a Multi-Line Heading or Footing

In the following free-form report, content is defined entirely in the sort heading, where text and data are stacked to support comparison among regions.

The first line of each subheading displays the name of the region. The next three lines display the following numbers decimally aligned. The decimal point displays one inch from the right of a two-inch column:

- Unit sales with no decimal places.
- Dollar sales with two decimal places.
- Budgeted sales with five decimal places.

```
DEFINE FILE GGSALES
 JUNITS/D15.0 = UNITS;
 JDOLLARS/D15.2 = DOLLARS;
 JBUDDOLLARS/D15.5 = BUDDOLLARS;
END
SET SQUEEZE = ON
TABLE FILE GGSALES
 BY REGION NOPRINT SUBHEAD
 "SALES FOR: <REGION "
 " "
 "Units: <JUNITS"
 "Dollar Sales: <JDOLLARS"
 "Budgeted Sales: <JBUDDOLLARS"
 ON TABLE SET PAGE-NUM OFF
 ON TABLE HOLD FORMAT HTML
 ON TABLE SET HTMLCSS ON
 ON TABLE SET STYLE *
 TYPE = REPORT, GRID = OFF ,$
 TYPE = SUBHEAD, ITEM = 1, WIDTH = 1.5                    ,$
 TYPE = SUBHEAD, ITEM = 2, WIDTH = 2, JUSTIFY=DECIMAL(1) ,$
 ENDSTYLE
END
```

In order for the second item to align, the first item must have a WIDTH specification so that it will take up a set amount of space. The rightmost character of the region name aligns over the digit to the left of the decimal point in the numeric values.

The output is:

```
SALES FOR:                    Midwest

Units:                        905,045.
Dollar Sales:              11,400,665.00
Budgeted Sales:            11,194,373.00000

SALES FOR:                   Northeast

Units:                        916,675.
Dollar Sales:              11,392,310.00
Budgeted Sales:            11,576,932.00000

SALES FOR:                   Southeast

Units:                        935,232.
Dollar Sales:              11,710,379.00
Budgeted Sales:            11,807,981.00000

SALES FOR:                        West

Units:                        932,039.
Dollar Sales:              11,652,957.00
Budgeted Sales:            11,641,513.00000
```

Procedure: How to Measure for Column Width and Decimal Alignment

Measuring Width. Determining the width of a heading or footing item is a three-step process:

1. Identify the maximum number of characters in a text string or field.
2. For a text string, count the characters. For a field, refer to the format specification in the Master File or in a command such as a DEFINE.
3. Measure the physical space in units (for example, inches) that is required to display the number of characters identified in step 1, based on the size of the font you are using. For example, the following value of the COUNTRY field would measure as follows:

Font	Font size	Comparison	Inches
Helvetica	10	England	.5
Times New Roman	10	England	.44
Courier	10	England	.56

Tip: Consider using a consistent set of fonts in your reports to make your measurements reusable.

Measuring for Decimal Alignment. After you have determined the width of an item, you can do a related measurement to determine the physical space required to display decimal data with a varying number of digits to the right of the decimal point.

1. Determine the maximum number of decimal places you need to accommodate to the right of the decimal place, plus the decimal point itself.
2. Measure the physical space in units (for example, inches) that is required to display the number of characters identified in step 1, based on the size of the font you are using.

Using Fields to Define Formats

How to:

Define and Apply a Format Field

Field-based reformatting allows you to apply different formats to each row in a single report column by using a field to identify the format that applies to each row. For example, you can use this technique to apply the appropriate decimal currency formats when each row represents a different country.

The field that contains the format specifications can be a:

- Real field in the data source.
- Temporary field created with a DEFINE command.
- DEFINE in the Master File.
- COMPUTE command. If the field is created with a COMPUTE command, the command must appear in the request prior to using the calculated field for reformatting.

The field that contains the formats must be alphanumeric and be at least eight characters in length. Only the first eight characters are used for formatting.

The field-based format may specify a length longer than the length of the original field. However, if the new length is more than one-third larger than the original length, the report column width may not be large enough to hold the value (indicated by asterisks in the field).

You can apply a field-based format to any type of field. However, the new format must be compatible with the original format:

- A numeric field can be reformatted to any other numeric format with any edit format options.
- An alphanumeric field can be reformatted to a different length.
- Any date field can be reformatted to any other date format type.
- Any date-time field can be reformatted to any other date-time format.

If the field-based format is invalid or specifies an impermissible type conversion, the field displays with plus signs (++++) on the report output.

For an illustration of this technique, see *Displaying Extended Currency Symbols* on page 1-123.

Syntax: How to Define and Apply a Format Field

- In a DEFINE command:

```
DEFINE FILE filename
 format_field/A8 = expression;
END
```

- In a Master File:

```
DEFINE format_field/A8 = expression; $
```

- In a request:

```
COMPUTE format_field/A8 = expression;
```

where:

format_field

Is the name of the field that contains the format for each row.

expression

Is the expression that assigns the format values to the format field.

Once the format field is defined, you can apply it in a report request:

```
TABLE FILE filename
 display fieldname/format_field[/just]
END
```

where:

display

Is any valid display command.

fieldname

Is a field in the request to be reformatted.

format_field

Is the name of the field that contains the formats. If the name of the format field is the same as an explicit format, the explicit format will be used. For example, a field named I8 cannot be used for field-based reformatting because it will be interpreted as the explicit format I8.

just

Is a justification option, L, R, or C. The justification option can be placed before or after the format field, separated from the format by a slash.

Punctuating Numbers

How to:

Determine the Punctuation of Large Numbers

Example:

Determining the Punctuation of Large Numbers

Countries differ in how they punctuate numbers; you can reflect these differences in reports using Continental Decimal Notation (CDN), specified with the SET CDN command. This command allows you to choose how to punctuate numbers, using a combination of commas, decimals, spaces, and single quotation marks.

You can use the SET CDN command in a report request but not in a DEFINE or COMPUTE command.

The punctuation specified by the SET CDN command also determines the punctuation used in numbers affected by the SET CENT-ZERO command. For syntax and an example of SET CENT-ZERO, see *Displaying a Leading Zero in Decimal Numbers* on page 1-119.

Syntax: How to Determine the Punctuation of Large Numbers

```
SET CDN = option
```

where:

option

Determines the punctuation used in numerical notation. Possible values are:

ON enables CDN. For example, the number 3,045,000.76 is represented as 3.045.000,76.

OFF disables CDN. For example, the number 3,045,000.76 is represented as 3,045,000.76. OFF is the default value.

SPACE separates groups of three significant digits with a space instead of a comma, and marks a decimal position with a comma instead of a period. For example, the number 3,045,000.76 is represented as 3 045 000,76.

QUOTE separates groups of three significant digits with a single quotation mark instead of a comma, and marks a decimal position with a comma instead of a period. For example, the number 3,045,000.76 is represented as 3’045’000,76.

Example: Determining the Punctuation of Large Numbers

In the following request, the CDN setting is ON, which punctuates numbers using a period to separate thousands, and a comma to separate decimals.

```
SET CDN = ON
TABLE FILE EMPLOYEE
 PRINT LAST_NAME FIRST_NAME SALARY
END
```

The output is:

LAST_NAME	FIRST_NAME	SALARY
STEVENS	ALFRED	$11.000,00
SMITH	MARY	$13.200,00
JONES	DIANE	$18.480,00
JONES	DIANE	$17.750,00
BANNING	JOHN	$29.700,00
IRVING	JOAN	$26.862,00
IRVING	JOAN	$24.420,00
ROMANS	ANTHONY	$21.120,00
MCCOY	JOHN	$18.480,00
BLACKWOOD	ROSEMARIE	$21.780,00
MCKNIGHT	ROGER	$16.100,00
MCKNIGHT	ROGER	$15.000,00
CROSS	BARBARA	$27.062,00
CROSS	BARBARA	$25.775,00

Displaying a Leading Zero in Decimal Numbers

How to:

Display a Leading Zero in Decimal-only Numbers

Example:

Displaying a Leading Zero in Decimal-only Numbers

By default, when a number containing only a decimal portion appears in a report, a leading zero is not displayed with the number. You can use the SET CENT-ZERO command to display a leading zero in decimal-only numbers.

Note that the CDN setting determines whether a decimal point or comma is the decimal separator. For information about the CND parameter, see *Punctuating Numbers* on page 1-118.

Syntax: How to Display a Leading Zero in Decimal-only Numbers

```
SET CENT-ZERO = {ON|OFF}
```

where:

`ON`

Displays a leading zero in decimal-only numbers.

`OFF`

Suppresses a leading zero in decimal-only numbers. OFF is the default value.

Example: Displaying a Leading Zero in Decimal-only Numbers

In the following request, the CENT-ZERO setting is ON, which displays a leading zero in decimal-only numbers:

```
SET CENT-ZERO = ON
SET PAGE-NUM = OFF
TABLE FILE CENTINV
 PRINT PRODNAME
 COMPUTE FACTOR = COST/RETAIL;
 BY PRODCAT
 WHERE PRODCAT EQ 'Cameras';
 ON TABLE SET STYLE *
 TYPE = REPORT, GRID = OFF ,$
 ENDSTYLE
END
```

The output is:

PRODCAT	Product Name:	FACTOR
Cameras	340SX Digital Camera 65K P	0.80
	330DX Digital Camera 1024K P	0.71
	AR3 35MM Camera 10 X	0.74
	AR2 35MM Camera 8 X	0.72

Selecting a Currency Symbol for Display

How to:

Display an Extended Currency Symbol

Example:

Displaying Extended Currency Symbols

You can select a currency symbol for display in report output regardless of the default currency symbol configured for National Language Support (NLS). Use the extended currency symbol format in place of the floating dollar (M) or non-floating dollar (N) display option.

When you use the floating dollar (M) or non-floating dollar (N) display option, the currency symbol associated with the default code page is displayed. For example, when you use an American English code page, the dollar sign is displayed.

The extended currency symbol format allows you to display a symbol other than the dollar sign. Using the extended currency symbol format, you can display the symbol for a United States dollar, a British pound, a Japanese yen, or the euro.

The extended currency support symbol formats are available for numeric formats (I, D, F, and P).

Syntax: How to Display an Extended Currency Symbol

numeric_format!*option*

where:

numeric_format

Is a valid numeric format (data types I, D, F, or P).

!

Is required.

option

Determines the currency symbol that is displayed, and whether the symbol is floating or non-floating. Possible values are:

d displays a non-floating dollar sign.

D displays a floating dollar sign.

e displays a non-floating euro symbol.

E displays a floating euro symbol.

l displays a non-floating British pound symbol.

L displays a floating British pound symbol.

y displays a non-floating Japanese yen symbol.

Y displays a floating Japanese yen symbol.

Example: Displaying Extended Currency Symbols

The following request displays the British pound on the row that represents England, the euro on the row that represents Italy, and the Japanese yen on the row that represents Japan.

```
DEFINE FILE CAR
 CFORMAT/A8 = DECODE COUNTRY('ENGLAND' 'F12.1C!L' 'JAPAN' 'D12!Y'
  ELSE 'D12.2!E');
END
TABLE FILE CAR
 PRINT SALES/CFORMAT DEALER_COST/CFORMAT
 BY COUNTRY
 WHERE COUNTRY EQ 'ENGLAND' OR 'JAPAN' OR 'ITALY'
 WHERE SALES GT 0
END
```

The output is:

COUNTRY	SALES	DEALER COST
ENGLAND	£12,000.0	£11,194.0
ITALY	€4,800.00	€4,915.00
	€12,400.00	€5,660.00
	€13,000.00	€5,660.00
JAPAN	¥43,000	¥2,626
	¥35,030	¥2,886

Displaying a Minus Sign After Negative Data

The Minus Edit display option enables you to display a minus sign to the right of negative numeric data. You can define this option in a Master File, or in a COMPUTE or DEFINE command.

Note that Minus Edit does not perform calculations intended to derive a negative number in the data. It functions only as a format option.

Options for displaying numeric data are fully documented in *Numeric Display Options* in the *Describing an Individual Field* chapter in the *Describing Data With WebFOCUS Language* manual.

Example: Describing the Minus Edit Format Option

In the Master File:

```
FIELDNAME = WHOLESALEPR, WPRC, F6.2- ,$
```

In a request:

```
DEFINE FILE MOVIES
 MDISC/D7.2 = 7.5;
END
TABLE FILE MOVIES
 SUM COMPUTE GROSS_PROFIT = LISTPR - WHOLESALEPR;
 NET_PROFIT/D12.2- = GROSS_PROFIT - MDISC;
 BY CATEGORY
 PRINT MDISC AS 'MEMBER,DISCOUNT'
 WHOLESALEPR AS 'WHOLESALE,PRICE'
 LISTPR AS 'LIST,PRICE'
 IF CATEGORY EQ 'ACTION' OR 'MUSICALS' OR 'DRAMA' OR 'SCI/FI';
 IF RECORDLIMIT EQ 10
END
```

The output is:

CATEGORY	GROSS PROFIT	NET PROFIT	MEMBER DISCOUNT	WHOLESALE PRICE	LIST PRICE
ACTION	8.96	1.46	7.50	10.99	19.95
DRAMA	9.98	2.48	7.50	10.00	19.98
MUSICALS	16.97	5.53-	7.50	13.99	19.98
			7.50	13.99	19.98
			7.50	9.99	14.98
SCI/FI	35.32	2.18-	7.50	14.99	19.98
			7.50	14.55	19.98
			7.50	13.99	19.95
			7.50	16.00	24.95
			7.50	19.99	29.98

Note: Results of the Minus Edit option can be seen in the NET_PROFIT column.

Displaying Compound Reports in PS Format

How to:

Display Compound Reports

Example:

Combining Report Formats and Graphs in a Compound Report

In addition to being able to combine multiple reports into a single PDF file, you can concatenate reports into a PS file using the SET COMPOUND command. The first PS or PDF report defines the format for the concatenated report, enabling you to intersperse intermediate reports of other formats (HTML, PDF, PS, EXL2K) into one encompassing report. You can then run or distribute the report with ReportCaster, which displays the compound PDF report in Adobe Acrobat Reader, or sends the compound PS report directly to a printer.

For a graphical implementation of this feature, see *Combining Multiple Reports Into a Single PDF or PS Report* in the *Saving and Reusing Report Output* chapter in the *Creating Reports With Graphical Tools* manual.

Syntax: How to Display Compound Reports

For a compound report that may contain different report types, use the syntax

```
SET COMPOUND = {OPEN|CLOSE} [NOBREAK]
```

or

```
ON TABLE SET COMPOUND {OPEN|CLOSE} [NOBREAK]
```

Note that when you are using this syntax, you must also include the following code to identify the display format of each of the reports to be concatenated:

```
ON TABLE PCHOLD FORMAT formatname
```

If all of the reports in the compound set are of the same type, either PDF or PS, you can use the following, more compact, syntax

```
ON TABLE PCHOLD FORMAT {PDF|PS} {OPEN|CLOSE} [NOBREAK]
```

where:

`OPEN`

Is specified with the first report, and begins the concatenation process. A report that contains the OPEN attribute must be in PDF or PS format.

`CLOSE`

Is specified with the last report, and ends the concatenation process.

NOBREAK

Is an optional phrase that suppresses page breaks. By default, each report is displayed on a separate page.

You can use NOBREAK selectively in a request to control which reports are displayed on the same page. Page numbering and headings are automatically supported with NOBREAK.

Example: Combining Report Formats and Graphs in a Compound Report

The following request generates a compound report from three different report types (PDF, HTML, and EXL2K), and embeds a graph in each report. Notice that each graph is saved as a GIF file in the graph request. The graph is then identified, sized, and positioned within the StyleSheet declaration (TYPE = REPORT, IMAGE = *graphname*…) of the report in which it is being embedded). Variations on the SET COMPOUND = syntax (OPEN, NOBREAK, CLOSE) combine the three reports on the same page. Key lines of code are highlighted in the following request.

Report 1:

```
GRAPH FILE SHORT
 SUM PROJECTED_RETURN AS 'Return on Investment'
 BY HOLDER
 ACROSS CONTINENT
 ON GRAPH SET LOOKGRAPH 3D_BAR
 ON GRAPH SET GRAPHEDIT SERVER
 ON GRAPH HOLD AS SLSGRPH1 FORMAT GIF
END
SET COMPOUND ='OPEN NOBREAK'
TABLE FILE SHORT
 SUM PROJECTED_RETURN AS 'Return on Investment'
 BY CONTINENT
 BY HOLDER
 HEADING
 "Investment Report"
 " "
 ON TABLE SET STYLE *
 TYPE = DATA, BACKCOLOR = (BY=B2 'SILVER' 'WHITE') ,$
 TYPE = HEADING, SIZE = 14, STYLE = BOLD ,$
 TYPE = REPORT, IMAGE = SLSGRPH1.gif, POSITION=(4.5 0.5),SIZE=(3.5 2.5),$
 ENDSTYLE
 ON TABLE PCHOLD FORMAT PS
END
```

Report 2:

```
GRAPH FILE TRADES
 SUM AMOUNT
 BY CONTINENT
 ON GRAPH SET LOOKGRAPH PIE
 ON GRAPH SET GRAPHEDIT SERVER
 ON GRAPH HOLD AS TRDSGR1 FORMAT GIF
END
SET COMPOUND = NOBREAK
TABLE FILE TRADES
 SUM AMOUNT AS 'Amount'
 BY CONTINENT AS 'Continent'
 BY REGION AS 'Region'
 HEADING
 "Trades Report"
 " "
 ON TABLE SET HTMLCSS ON
 ON TABLE SET STYLE *
 TYPE = DATA, BACKCOLOR = (BY = B2 'SILVER' 'WHITE') ,$
 TYPE = HEADING, SIZE = 14, STYLE = BOLD ,$
 TYPE = REPORT, IMAGE = TRDSGR1.gif, POSITION = (4 3), SIZE = (4 2.5) ,$
 ENDSTYLE
 ON TABLE PCHOLD FORMAT HTML
END
```

Report 3:

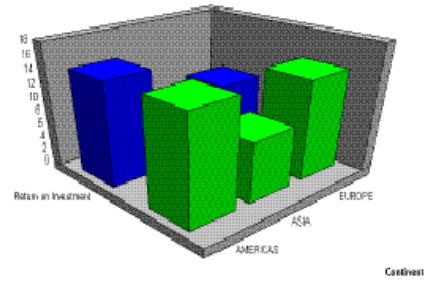
```
GRAPH FILE SHORT
 SUM BALANCE
 BY CONTINENT
 ON GRAPH SET LOOKGRAPH 3D_BAR
 ON GRAPH SET GRAPHEDIT SERVER
 ON GRAPH SET STYLE *
 TYPE = DATA, COLOR = RED ,$
 ENDSTYLE
 ON GRAPH HOLD AS BALGR1 FORMAT GIF
END
SET COMPOUND = CLOSE
TABLE FILE SHORT
 SUM BALANCE AS 'Balance'
 BY CONTINENT AS 'Continent'
 BY REGION AS 'Region'
 HEADING
 "Balance by Region"
 " "
 ON TABLE SET STYLE *
 TYPE = DATA, BACKCOLOR = (BY = B2 'SILVER' 'WHITE') ,$
 TYPE = HEADING, SIZE = 14, STYLE = BOLD ,$
 TYPE = REPORT, IMAGE = BALGR1.gif, POSITION = (4 6), SIZE = (4 2.5) ,$
 ENDSTYLE
 ON TABLE PCHOLD FORMAT EXL2K
END
```

The output is:

PAGE 1

Investment Report

Continent	Instrument Holder	Return on Investment
AMERICAS	COMM	14.980
	GOVT	15.400
ASIA	COMM	8.400
	GOVT	8.680
EUROPE	COMM	9.850
	GOVT	14.280

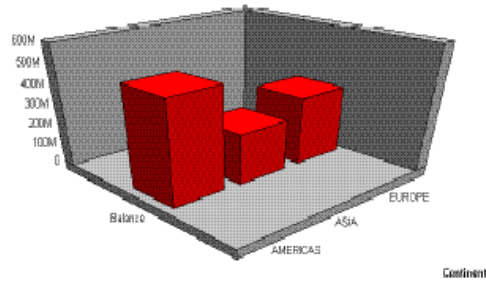


Trades Report

Continent	Region	Amount
AMERICAS	CENTRAL AMERICA	451,331,300
	NORTH AMERICA	1,370,452,600
	SOUTH AMERICA	454,334,100
ASIA	FAR EAST	954,384,400
	MID EAST	230,438,500
EUROPE	EAST EUROPE	436,090,610
	WEST EUROPE	1,042,079,900

Balance by Region

Continent	Region	Balance
AMERICAS	CENTRAL AMERICA	40,200,667
	NORTH AMERICA	367,610,683
	SOUTH AMERICA	87,661,381
ASIA	FAR EAST	202,660,842
	MIDDLE EAST	44,659,771
EUROPE	EASTERN EUROPE	121,977,597
	WESTERN EUROPE	210,550,374



Excel Integration Enhancements

How to:

Save Reports as FORMAT EXL2K FORMULA

Example:

Displaying Formatted Dates in Excel 2000

Generating Native Excel Formulas for Column Totals

Customizing Worksheet Tabs in EXL2K Reports

Viewing a Report in EXL97 Format

The Excel 2000 format (EXL2K) has been enhanced to:

- **Display Formatted Dates and Numeric Values in Excel 2000.** Excel 2000 spreadsheets generated by the server contain the same numeric formatting as specified in the data source's Master File or as specified in a temporary field. Numeric values and date formats (such as currency and Smart Dates) are translated to supported Excel formats and are displayed properly in Excel 2000. For an example, see *Displaying Formatted Dates in Excel 2000* on page 1-131.
- **Generate Native Excel Formulas in Excel 2000.** When you display or save a report request using EXL2K FORMULA, the resulting spreadsheet contains an Excel formula that computes and displays the results of any type of summed information (such as column totals, row totals, subtotals, and calculated values), rather than static numbers. Spreadsheets saved using the EXL2K FORMULA format are interactive, allowing for "what if" scenarios that immediately reflect any additions or modifications made to the data. For an example, see *Generating Native Excel Formulas for Column Totals* on page 1-133.
- **Customize Worksheet Tabs.** By default, when you choose EXL2K as your display format, the report opens in an Excel 2000 worksheet, identified in a tab at the bottom of the spreadsheet as Sheet1, Sheet2, and so on. For an example, see *Customizing Worksheet Tabs in EXL2K Reports* on page 1-134.

  You can change the name of a Sheet tab to make it more descriptive by including the TITLETEXT attribute in the StyleSheet declaration.

  Note that the same syntax changes the browser title bar in an HTML report.
- **View and Save a Report in Excel 97 Format.** The EXL97 format allows you to view and save reports in Excel 97. When specifying the EXL97 format, an HTML-based file is generated with an extension of .e97. The appropriate MIME type is automatically assigned to designate Excel as the active application for this file type. For an example, see *Viewing a Report in EXL97 Format* on page 1-135.

  Format EXL97 is fully compatible with Excel 2000 and Excel 2002.

For a graphical implementation of these enhancements, see *Saving a Report as Excel 2000, Excel 97, and Excel Output* in the *Saving and Reusing Report Output* chapter in the *Creating Reports With Graphical Tools* manual.

Syntax: How to Save Reports as FORMAT EXL2K FORMULA

Add the following syntax to your request to take advantage of Excel formulas in your spreadsheet:

```
ON TABLE {PCHOLD|HOLD} FORMAT EXL2K FORMULA
```

where:

`PCHOLD`

Displays the output in an Excel 2000 spreadsheet.

`HOLD`

Saves the output for reuse in an Excel 2000 spreadsheet.

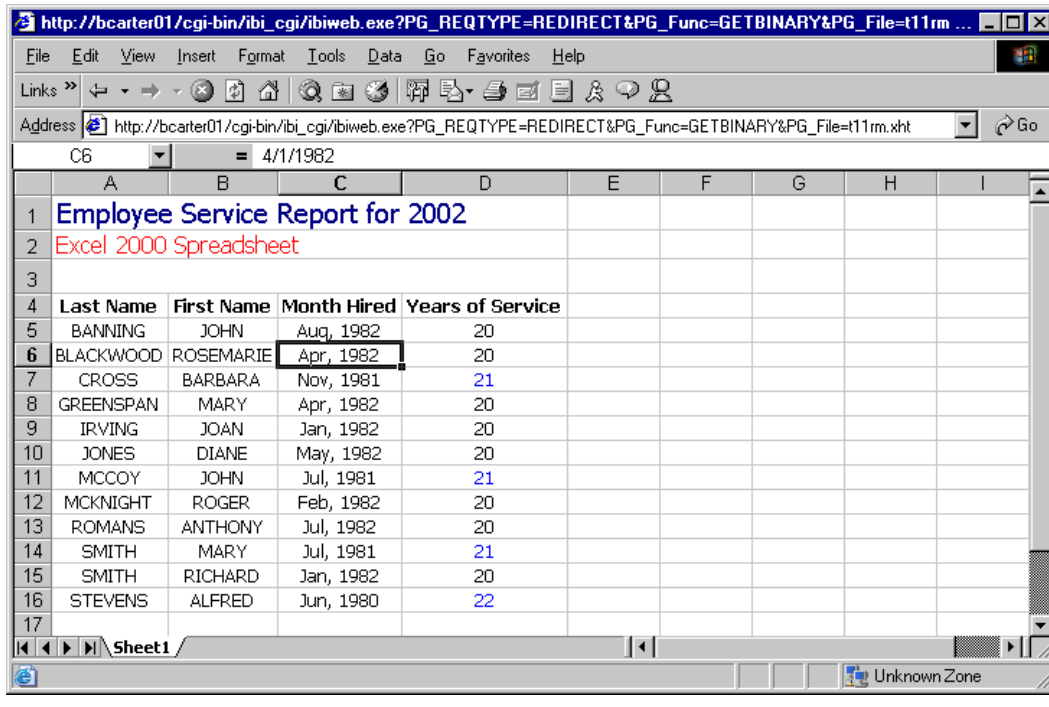
Example: Displaying Formatted Dates in Excel 2000

The following request illustrates how customized dates appear in a spreadsheet when you use the EXL2K format.

- The format for Month Hired is defined in the request as MtYY (the month is represented as a 3-character abbreviation with an initial capital letter followed by a four-digit year).
- The format for Years of Service is defined as I4C, a four digit integer with a comma if required. Both formats are properly displayed as defined in the spreadsheet.

```
SET PAGE-NUM = OFF
DEFINE FILE EMPLOYEE
 YRHIRED/YY = HIRE_DATE;
 MHIRED/MtYY = HIRE_DATE;
 TOTSVC/I4C = 2002 - YRHIRED;
END
TABLE FILE EMPLOYEE
 "Employee Service Report for 2002"
 "Excel 2000 Spreadsheet"
 " "
 PRINT FIRST_NAME AS 'First Name'
 MHIRED AS 'Month Hired'
 TOTSVC AS 'Years of Service'
 BY LAST_NAME AS 'Last Name'
 ON TABLE SET BYDISPLAY ON
 ON TABLE PCHOLD FORMAT EXL2K
 ON TABLE SET STYLE *
 TYPE = REPORT, GRID = OFF, FONT = TAHOMA ,$
 TYPE = HEADING, SIZE = 14, COLOR = NAVY ,$
 TYPE = HEADING, LINE = 2, SIZE = 12, COLOR = RED ,$
 TYPE = TITLE, JUSTIFY = CENTER, STYLE = BOLD ,$
 TYPE = DATA, JUSTIFY = CENTER ,$
 TYPE = DATA, COLUMN = TOTSVC, COLOR = BLUE, WHEN = TOTSVC GT 20 ,$
 ENDSTYLE
END
```

The output is:


http://bcarter01/cgi-bin/ibi_cgi/ibiweb.exe?PG_REQTYPE=REDIRECT&PG_Func=GETBINARY&PG_File=t11rm ...
File Edit View Insert Format Tools Data Go Favorites Help
Links
Address http://bcarter01/cgi-bin/ibi_cgi/ibiweb.exe?PG_REQTYPE=REDIRECT&PG_Func=GETBINARY&PG_File=t11rm.xht
Go
C6 = 4/1/1982
Employee Service Report for 2002
Excel 2000 Spreadsheet
Last Name First Name Month Hired Years of Service
BANNING JOHN Aug, 1982 20
BLACKWOOD ROSEMARIE Apr, 1982 20
CROSS BARBARA Nov, 1981 21
GREENSPAN MARY Apr, 1982 20
IRVING JOAN Jan, 1982 20
JONES DIANE May, 1982 20
MCCOY JOHN Jul, 1981 21
MCKNIGHT ROGER Feb, 1982 20
ROMANS ANTHONY Jul, 1982 20
SMITH MARY Jul, 1981 21
SMITH RICHARD Jan, 1982 20
STEVENS ALFRED Jun, 1980 22
Sheet1
Unknown Zone

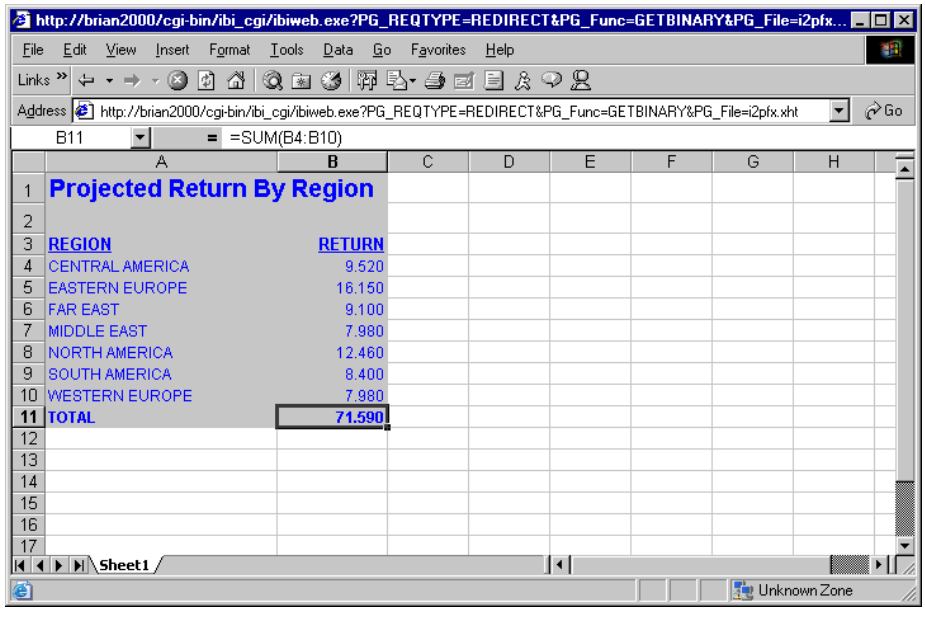
Example: Generating Native Excel Formulas for Column Totals

The following request illustrates how a column total in a report request is translated to an Excel formula when you use the format EXL2K FORMULA. Notice that the formatting of the column total (TYPE = GRANDTOTAL) is retained in the Excel 2000 spreadsheet.

When you select the total in the report, the equation =SUM(B4:B10) displays in the formula bar, representing the column total as a sum of cell ranges.

```
TABLE FILE SHORT
 HEADING
 "Projected Return By Region"
 " "
 SUM PROJECTED_RETURN AS 'RETURN'
 BY REGION AS 'REGION'
 ON TABLE COLUMN-TOTAL
 ON TABLE SET STYLE *
 TYPE = REPORT, COLOR = BLUE, BACKCOLOR = SILVER, SIZE = 9 ,$
 TYPE = HEADING, STYLE = BOLD, SIZE = 14 ,$
 TYPE = TITLE, STYLE = BOLD+UNDERLINE, SIZE = 10 ,$
 TYPE = GRANDTOTAL, STYLE = BOLD ,$
 ENDSTYLE
 ON TABLE PCHOLD FORMAT EXL2K FORMULA
END
```

The output is:



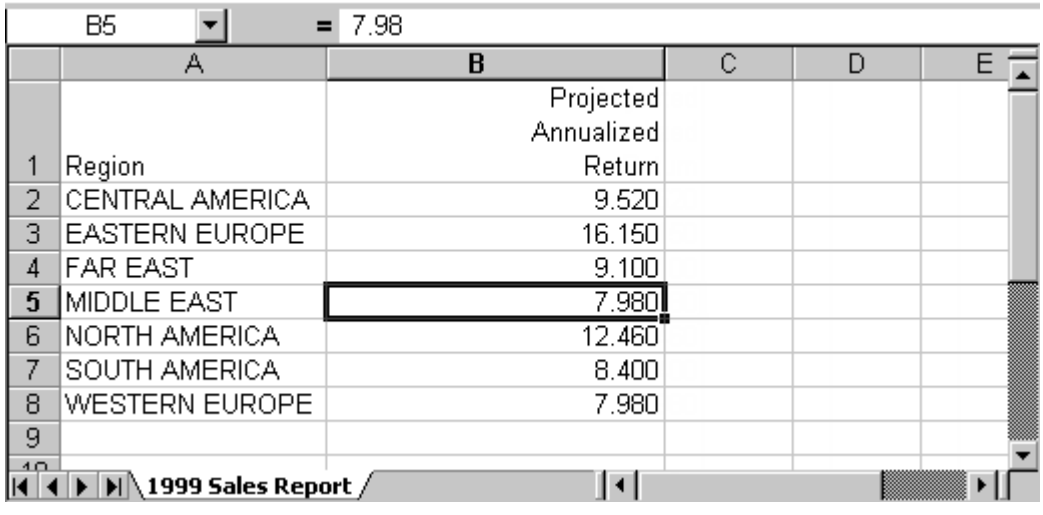
You can translate any total (subtotal, row total, or column total) to an Excel formula.

Example: Customizing Worksheet Tabs in EXL2K Reports

The following request illustrates how you can replace the default worksheet tab name in an EXL2K report using the TITLETEXT attribute in your StyleSheet.

```
TABLE FILE SHORT
 SUM PROJECTED_RETURN
 BY REGION
 ON TABLE PCHOLD FORMAT EXL2K
 ON TABLE SET STYLE *
 TYPE = REPORT, TITLETEXT = '1999 Sales Report' ,$
 ENDSTYLE
END
```

The output is:

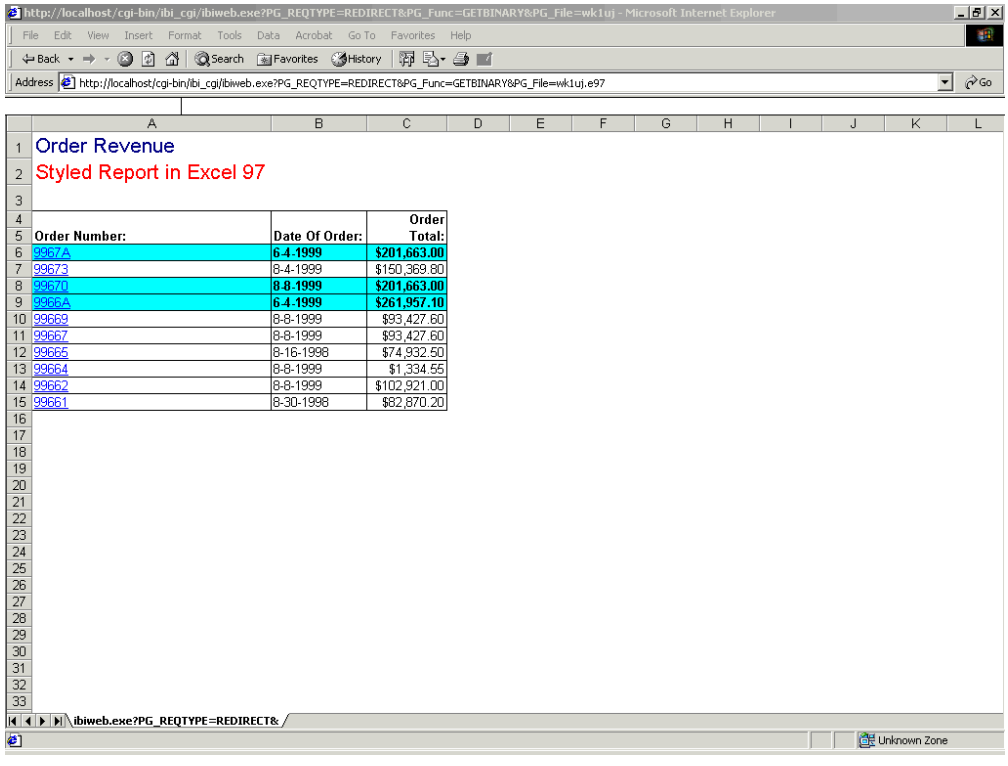


Example: Viewing a Report in EXL97 Format

The following request illustrates how a report appears when you use the PCHOLD FORMAT EXL97 command. The request includes drill-downs to another procedure and formatting options, including conditional styling.

```
TABLE FILE CENTORD
 HEADING
 "Order Revenue"
 "Styled Report in Excel 97"
 " "
 SUM ORDER_DATE LINEPRICE AS 'Order,Total:'
 BY HIGHEST 10 ORDER_NUM
 ON TABLE SET PAGE-NUM OFF
 ON TABLE SET ACCESSIBLE 508
 ON TABLE PCHOLD FORMAT EXL97
 ON TABLE SET STYLE *
 TYPE = HEADING, COLOR = NAVY, SIZE = 14 ,$
 TYPE = HEADING, LINE = 2, COLOR = RED ,$
 TYPE = DATA,BACKCOLOR = AQUA,STYLE = BOLD,WHEN = LINEPRICE GT 200000 ,$
 TYPE = TITLE, STYLE = BOLD ,$
 TYPE = DATA, COLUMN = ORDER_NUM, FOCEXEC = DETAILS ,$
 ENDSTYLE
END
```

The output is:



Note: Although format EXL97 is available, there are some limitations when compared to the Excel 2000 (EXL2K) formats. Future enhancements in the area of Excel integration will primarily be made to the EXL2K formats. We recommend upgrading to Excel 2000 or higher so you can take full advantage of our Excel integration, as well as all future enhancements.

Selecting a Language for Excel Requests

In this section:

Setting the Default Column Width for Excel

How to:

Set the Language for Excel Requests

You can include the SET EXL2KLANG command in the nlscfg.err file to specify the language you wish to use for Microsoft® Excel requests. This language must be the same as the language of Excel on the browser machine in order to correctly display output.

You can add the SET EXL2KLANG command in a profile or procedure to override the setting in the errors file.

Note: You must specify a column width that is valid for the language supplied for the SET EXL2KLANG command. See *Setting the Default Column Width for Excel* on page 1-138.

Syntax: How to Set the Language for Excel Requests

You can code the SET EXL2KLANG command in a profile or procedure to override the setting in the errors file.

```
SET EXL2KLANG = {language|ENG}
```

where:

language

Is the Excel language. Possible values are:

ENG for English. ENG is the default value.

FRE for French.

GER for German.

JPN for Japanese.

KOR for Korean.

SPA for Spanish.

Setting the Default Column Width for Excel

How to:

Set the Excel Default Column Width in nlscfg.err

Example:

Specifying German With Column Width 10

The EXL2K_DEFCOLW parameter in nlscfg.err sets the default column width for the language version of Excel running on the WebFOCUS Reporting Server.

For English or Japanese, 8 is the default value. For Western European languages, 10 is the default value.

Syntax: How to Set the Excel Default Column Width in nlscfg.err

You must specify the default column width that is valid for the language specified for EXL2KLANG.

```
EXL2K_DEFCOLW = nn
```

where:

nn

Is the default column width for Excel.

Example: Specifying German With Column Width 10

To run the WebFOCUS Reporting Server using German for Excel, code the following in the nlscfg.err file:

```
EXL2KLANG = GER
EXL2K_DEFCOLW = 10
```

Stacking Columns With Multiple FOLD-LINEs

How to:

Stack Columns With Multiple FOLD-LINE Phrases

Example:

Stacking Columns With Multiple FOLD-LINE Phrases

You can reduce column width by including up to 16 FOLD-LINE phrases in a report request. Previously, you could only include one per request.

FOLD-LINE is available for PDF and PS reports, but not for cell-based reports such as HTML and Excel.

Syntax: How to Stack Columns With Multiple FOLD-LINE Phrases

```
display fieldname1... FOLD-LINE fieldname2... {ON|BY} fieldname3 FOLD-LINE
```

where:

display

Is any display command.

fieldname2, fieldname3, ...

Are field names that follow FOLD-LINE phrases are moved to a new line.

The field name may be a sort field or display field. If it is a display field, the next field is placed at the start of the next line. If it is a sort field, the next field is on a new line, offset by two spaces from the beginning of the current line.

Example: Stacking Columns With Multiple FOLD-LINE Phrases

```
SET ONLINE-FMT = PDF
TABLE FILE MOVIES
 SUM WHOLESALEPR LISTPR
 BY CATEGORY FOLD-LINE
 BY TITLE
 BY DIRECTOR FOLD-LINE
 IF CATEGORY EQ 'FOREIGN' OR 'DRAMA'
END
```

The output is:

```
CATEGORY
   TITLE                                DIRECTOR            WHOLESALEPR
   -----                                --------            -----------

DRAMA
   DOG DAY AFTERNOON                    LUMET S.              10.00
FOREIGN
   BABETTE'S FEAST                      AXEL G.               13.25

   DAS BOOT                             PETERSON W.           15.99

   DEATH IN VENICE                      VISCONTI L.           30.00

   FAMILY, THE                          SCOLA E.              32.00

   FANNY AND ALEXANDER                  BERGMAN I.            25.00

   TIN DRUM, THE                        SCHLONDORFF V.        30.00
```

Specifying Multiple Drill-Down Links

You can create multiple drill-down links for a single value on a summary report, chosen from any combination of supported actions. For example, you can create links to a detail report or Maintain procedure, a JavaScript function, and a URL. The user selects the desired drill-down action from a pop-up menu that is automatically generated.

This feature is available for HTML reports and format HTMTABLE. Developer Studio has an interface that automatically generates the code. See *Creating Multiple Drill Downs With the Report Painter* in Chapter 2, *Developer Studio Enhancements*.

The types of links you can create are fully documented in the *Linking a Report to Other Resources* chapter in the *Creating Reports With WebFOCUS Language* manual.

Syntax: How to Create Multiple Drill-Down Links

```
TYPE = type, [subtype,] DRILLMENUITEM = '{description'|'DrillDown n'},
 type_of_link
```

where:

type

Identifies the report component that you select in the Web browser to execute the link. The TYPE attribute and its value must appear at the beginning of the declaration.

subtype

Are any additional attributes, such as COLUMN, LINE, or ITEM, that are needed to identify the report component that you are formatting.

description

Is the text that appears on the pop-up menu of drill-down options on the report output. DrillDown *n* is the default value, where *n* is a consecutive integer, such as DrillDown 1, DrillDown 2, and so on.

type_of_link

Is the type of link; for example, a link to another report, a URL, a URL from a field, a JavaScript function, a WebFOCUS Maintain procedure, or a WebFOCUS compiled Maintain procedure.

Navigating Reports From a Table of Contents

In this section:

Grouping Sort Fields for Display

How to:

Add a Multi-Level TOC as a Report Icon

Add TOCs to a Heading

Example:

Navigating Sorted Data From a TOC Icon

Navigating a Multi-Level HTML TOC in a Page Heading

You can enhance navigation within a large HTML report by adding a dynamic HTML-based Table of Contents (TOC) for multiple sort fields. Enabling a TOC for a lower-level sort field automatically enables a TOC for all parent sort fields. The TOCs display, as hyperlinks, all values of the first (highest-level) vertical sort field, as well as the values of any lower-level BY fields that you designate for inclusion.

The TOC can appear as an expandable icon in the upper left corner of the report or as one or more drop-down lists in a page heading or footing or a report heading or footing.

If you select a value in the TOC, that value flashes (it is highlighted in gray) to draw your attention to it in the browser window. Where the flash appears and whether and how the screen display changes is controlled by the following factors:

- When you change the highest level sort group from the TOC (either from the hierarchy above the report or from the first drop-down list in a heading or footing), the selected value flashes three times in the browser window.
- When you change a lower-level sort value within the current high-level sort group, the selected value flashes three times in the window. This is because you are still within the same major sort group, and, therefore, within the same page break. From the selected value at the top of the window, you can then scroll quickly to the related details.

  If the selected lower-level value is already viewable on the screen, and the remaining report will fit on the screen, the value flashes, but the report does not scroll.

The TOC also enhances the display of groups of data. You can view one section (or page) of the report at a time, or you can view all sections at once. You can control this with a page break. For more information, see *Grouping Sort Fields for Display* on page 1-147.

For a graphical implementation of this feature, see *Navigating Sort Groups From a Table of Contents* in the *Creating Reports With the Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.

Syntax: How to Add a Multi-Level TOC as a Report Icon

Using a SET command, the syntax is

```
SET COMPOUND = 'BYTOC [n]'
ON TABLE SET COMPOUND 'BYTOC [n]'
```

Using a PCHOLD command, the syntax is

```
ON TABLE PCHOLD FORMAT HTML BYTOC [n]
```

where:

n

Represents the number of vertical sort (BY) fields to include in the TOC, beginning with the first (highest-level) sort field in the request. The hierarchy of sort fields is determined by the order in which they are specified in the request.

1 is the default value, meaning that only the highest-level sort field and its values are displayed in the TOC.

By default, a section break is placed after the first (highest-level) sort field, unless otherwise specified in the request. For details, see *Grouping Sort Fields for Display* on page 1-147.

Note: Single quotation marks are required when you use the SET command to specify the number of sort fields you want to include in the TOC. Single quotation marks are not required when BYTOC is specified without a number or when it is specified in a PCHOLD command, with or without a number.

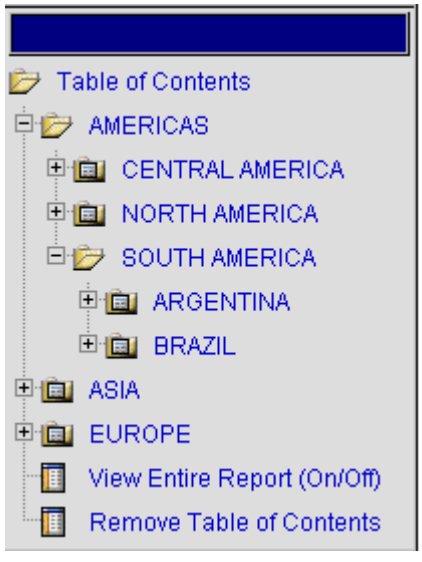
Example: Navigating Sorted Data From a TOC Icon

The following request uses the SET COMPOUND command to add a dynamic HTML Table of Contents as an icon in the upper left corner of the report. The TOC displays a hierarchy consisting of *five* levels of sort fields, beginning with the first (highest-level). The sort fields are: CONTINENT, REGION, COUNTRY, HOLDER, and TYPE.

```
TABLE FILE SHORT
 PRINT PROJECTED_RETURN
 BY CONTINENT
 BY REGION
 BY COUNTRY
 BY HOLDER
 BY TYPE
 ON TABLE SET PAGE-NUM OFF
 ON TABLE PCHOLD FORMAT HTML
 ON TABLE SET COMPOUND 'BYTOC 5'
 ON TABLE SET HTMLCSS ON
 ON TABLE SET STYLE *
 TYPE = REPORT, GRID = OFF, FONT = 'ARIAL', SIZE = 9, STYLE = NORMAL ,$
 ENDSTYLE
END
```

The output is displayed with the TOC object in the upper left corner. Double-click the object to expand the Table of Contents at the continent level:

Click the + sign next to Americas, then click the + sign next to South America.



The field values (Argentina and Brazil) are listed in the TOC. (These are values of the field COUNTRY. If you wish to see the field name of a value in the TOC, hover over that value with your cursor.)

Continue to navigate to the detail you want to view by choosing values at any sort level in the TOC.

- Clicking a + sign expands the field to display its values in the TOC.
- Clicking an actual value (hyperlink) in the TOC momentarily highlights that value and, if necessary, adjusts the report display to move the value into view. The TOC collapses to its icon when you click *Table of Contents*, but you can continue to scroll back, expand the TOC, and make additional selections.

Syntax: How to Add TOCs to a Heading

Include the following attribute in your StyleSheet declaration

```
TYPE = heading, [subtype,] TOC = sort_column ,$
```

where:

heading

Is the type of heading or footing that contains the TOC. Possible values are:

TABHEADING specifies the report heading.

TABFOOTING specifies the report footing.

HEADING specifies the page heading.

FOOTING specifies the page footing.

subtype

Are attributes that identify the location in the heading or footing where each requested drop-down list will be displayed. These options can be used separately or in combination, depending upon the degree of specificity required to identify a component. Possible values are:

- LINE_# identifies a line by its position in a heading or footing.

  If a heading or footing has multiple lines and you apply a StyleSheet declaration that does not specify LINE_#, the declaration is applied to *all* lines. Blank lines are counted when interpreting the value of LINE.

- LINE=*n* is required if a heading or footing has multiple lines; otherwise, you can omit it.

- OBJECT identifies the TOC object in a heading or footing as a text string or field value. Possible values are TEXT or FIELD.

  You can use a field and/or text as a placeholder for a TOC drop-down list; however, using a field is preferred. (If the TOC feature is not in effect, the field name is displayed in the report.)

  TEXT may represent free text or a Dialogue Manager amper (&) variable.

  It is not necessary to specify OBJECT = TEXT unless you are styling both text strings and embedded fields in the same heading or footing.

  For related information, see ITEM_#.

- ITEM_# identifies an item by its position in a line.

  To determine an ITEM_# for an OBJECT, follow these guidelines:

  When used with OBJECT = TEXT, count only the text strings from left to right.

  When used with OBJECT = FIELD, count only the fields from left to right.

  If you apply a StyleSheet declaration that specifies ITEM_#, the number is counted from the beginning of each line in the heading or footing, not just from the beginning of the first line.

sort_column

Identifies the vertical sort columns (BY fields) to include as TOCs. You can identify a column using the following notations: N*n*, P*n*, B*n*, or *fieldname*. For details about column notation in a StyleSheet, see *How to Identify an Entire Column* in the *Identifying a Report Component in WebFOCUS* chapter in the *Creating Reports With WebFOCUS Language* manual.

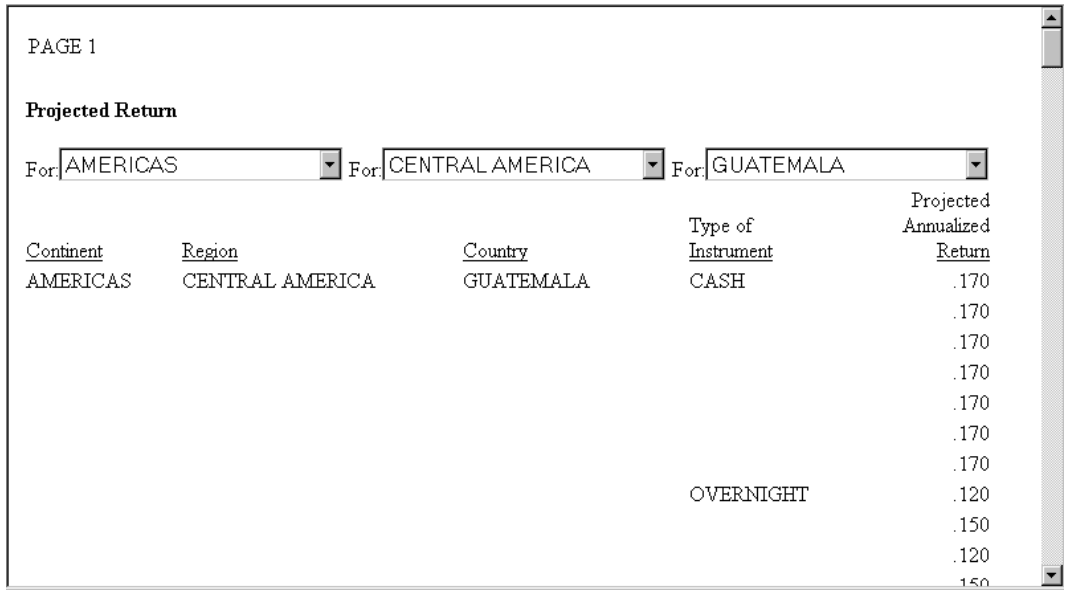
Note: You must maintain the hierarchy of BY fields because the TOC objects in headings (the drop-down lists) are interdependent and correspond with the hierarchy in the report.

Example: Navigating a Multi-Level HTML TOC in a Page Heading

The following request uses a StyleSheet to add an HTML TOC that contains drop-down lists in the third line of the page heading for three sort (BY) fields specified in the request: CONTINENT, REGION, and COUNTRY. Each field becomes a place-holder for its TOC. (If the TOC features were not in effect, the field would appear in the report.)

```
TABLE FILE SHORT
 "Projected Return"
 " "
 "For:<CONTINENT For:<REGION For:<COUNTRY"
 PRINT PROJECTED_RETURN
 BY CONTINENT BY REGION BY COUNTRY BY TYPE
 ON TABLE PCHOLD FORMAT HTML
 ON TABLE SET HTMLCSS ON
 ON TABLE SET STYLE *
 TYPE = REPORT, GRID = OFF ,$
 TYPE = HEADING, LINE = 1, STYLE = BOLD ,$
 TYPE = HEADING, LINE = 3, OBJECT = FIELD, ITEM = 1, TOC = 1 ,$
 TYPE = HEADING, LINE = 3, OBJECT = FIELD, ITEM = 2, TOC = 2 ,$
 TYPE = HEADING, LINE = 3, OBJECT = FIELD, ITEM = 3, TOC = 3 ,$
 ENDSTYLE
END
```

The output is:



PAGE 1

Projected Return

For: AMERICAS For: CENTRAL AMERICA For: GUATEMALA

Continent	Region	Country	Type of Instrument	Projected Annualized Return
AMERICAS	CENTRAL AMERICA	GUATEMALA	CASH	.170
				.170
				.170
				.170
				.170
				.170
				.170
			OVERNIGHT	.120
				.150
				.120
				.150

Grouping Sort Fields for Display

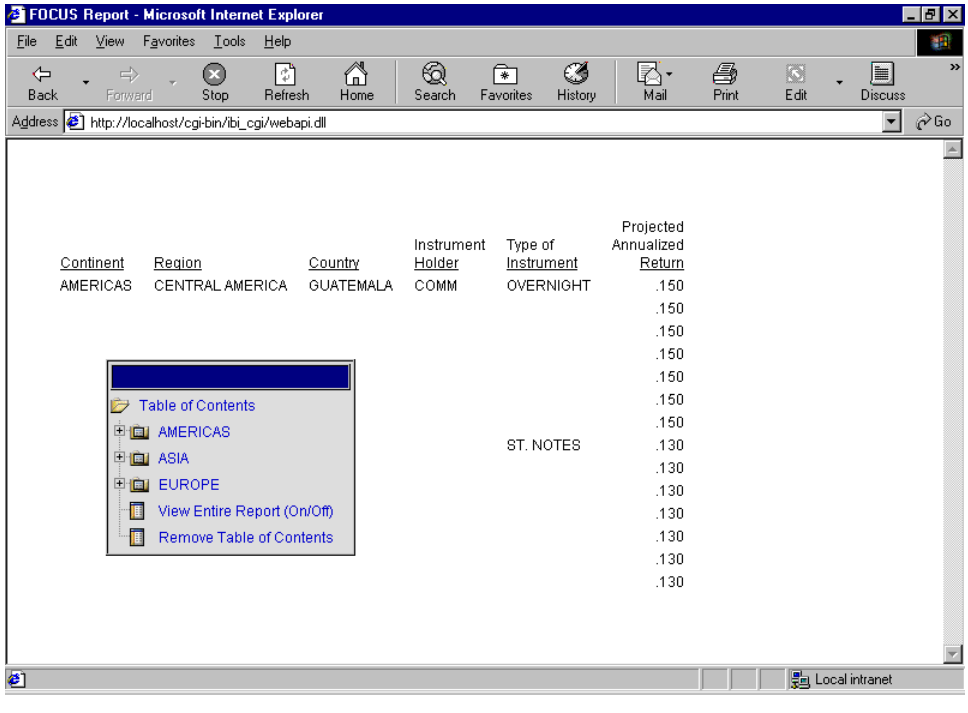
Data in a TOC report is grouped into sections based on the sort fields. TOC reports only display one section at a time for easier viewing. Each section contains all the values for its sort field. You can customize each section with a page break. By default, a page break is included in the first (highest level) sort field. You can add page breaks to create additional sections and group the data based on a lower-level sort field.

Example: Customizing Sections of the Report With a Page Break

```
TABLE FILE SHORT
 PRINT PROJECTED_RETURN
 BY CONTINENT
 BY REGION
 BY COUNTRY
 BY HOLDER
 BY TYPE
 ON HOLDER PAGE-BREAK
 ON TABLE SET PAGE-NUM OFF
 ON TABLE PCHOLD FORMAT HTML
 ON TABLE SET COMPOUND 'BYTOC 5'
 ON TABLE SET HTMLCSS ON
 ON TABLE SET STYLE *
 TYPE = REPORT, GRID = OFF, FONT = 'ARIAL', SIZE = 9, COLOR = 'BLACK',
  BACKCOLOR = 'NONE', STYLE = NORMAL ,$
 ENDSTYLE
END
```

One section of the report is displayed at a time.

The report is broken into sections based on the values for HOLDER. You will see the detail for each value of HOLDER in a single section.



Producing Row Totals for ACROSS Field Values

How to:

Produce Row Totals for Horizontal (ACROSS) Sort Field Values

Example:

Producing Row Totals for Horizontal (ACROSS) Sort Field Values

You can produce row totals for horizontal (ACROSS) sort field values. Row totals for horizontal sort fields are different from standard row totals in that only the horizontal sort field values are included in the total.

Syntax: How to Produce Row Totals for Horizontal (ACROSS) Sort Field Values

```
ACROSS sortfield ACROSS-TOTAL [AS 'name'] [COLUMNS col1 AND col2 ...]
```

where:

sortfield

Is the name of the field being sorted across.

name

Is the new name for the ACROSS-TOTAL column title.

col1, col2

Are the titles of the ACROSS columns you want to include in the total.

Example: Producing Row Totals for Horizontal (ACROSS) Sort Field Values

The following request illustrates how to generate a row total for horizontal (ACROSS) sort field values. Notice that the summed values in the TOTAL TITLE COUNT column only reflect the values in the PG RATING and R RATING columns. The values in the COPIES column are not included, since they are not horizontal (ACROSS) sort field values.

```
TABLE FILE MOVIES
 SUM COPIES BY CATEGORY
 COUNT TITLE BY CATEGORY
 ACROSS RATING ACROSS-TOTAL
 COLUMNS PG AND R
END
```

The output is:

		RATING		
		PG	R	TOTAL
		TITLE	TITLE	TITLE
CATEGORY	COPIES	COUNT	COUNT	COUNT
ACTION	14	2	3	5
COMEDY	16	4	1	5
DRAMA	2	0	1	1
FOREIGN	5	2	3	5
MUSICALS	2	1	1	2
MYSTERY	17	2	5	7
SCI/FI	3	0	3	3

Output Formats for Application Processing

In this section:

COM Format

COMT Format

TAB Format

TABT Format

How to:

Control Retrieval of Comma-delimited Data Sources With PCOMMA

Four output formats—COM, COMT, TAB, and TABT—enable you to save report output to a designated directory for use with a desktop product.

- Comma-delimited data sources are sequential data sources in which field values are separated by commas. This format type can be imported into applications, such as Excel or Lotus.
- Tab-delimited data sources are sequential data sources in which field values are separated by tabs.

Supported output formats are fully documented in *Choosing Output File Formats* in the *Saving and Reusing Your Report Output* chapter in the *Creating Reports With WebFOCUS Language* manual.

For graphical implementations of these formats, see *Comma-delimited and Tab-delimited Output Formats* in Chapter 2, *Developer Studio Enhancements*.

COM Format

The COM format saves the data values as a variable-length text file, with fields separated by commas and with character values enclosed in double quotation marks. This is useful for further processing in a database application. This format type can be imported into applications such as Excel or Lotus.

Leading blanks are removed from numeric fields, and trailing blanks are removed from character fields. Issuing a request against this data source requires the SET PCOMMA = ON command. See *Control Retrieval of Comma-delimited Data Sources With PCOMMA* on page 1-152.

The COM format includes a built-in safety feature that allows embedded quotation marks within text fields. It inserts a second double quotation mark (") adjacent to the existing one. For example, if you input Joe "Smitty" Smith, the output will be Joe ""Smitty"" Smith.

The extension or file type for this format is CSV. A Master File is created for this format type when the command used to create the output file is HOLD. The SUFFIX in the generated Master File is COM.

Syntax: How to Control Retrieval of Comma-delimited Data Sources With PCOMMA

```
SET PCOMMA = {ON|OFF}
```

where:

`ON`

Enables the retrieval of comma-delimited data sources created by a PC application. It indicates that alphanumeric data is enclosed in double quotation marks and each record is completely contained on one line and is terminated with a carriage return and line feed.

`OFF`

Does not enable the retrieval of comma-delimited data sources created by a PC application. It indicates that alphanumeric data is not enclosed in double quotation marks and each record is terminated with a comma and dollar sign. OFF is the default value.

COMT Format

The COMT format saves the column headings in the first row of the output file. It produces a variable-length text file, with fields separated by commas and with character values enclosed in double quotation marks. This is useful for further processing in a database application. This format type can be imported into applications such as Excel or Lotus.

Leading blanks are removed from numeric fields, and trailing blanks are removed from character fields. This format is required by certain software packages such as Microsoft Access.

The COMT format includes a built-in safety feature that allows embedded quotation marks within text fields. It inserts a second double quotation mark (") adjacent to the existing one. For example, if you input Joe "Smitty" Smith, the output will be Joe ""Smitty"" Smith.

The extension or file type for this format is CSV. A Master File is created for this format type when the command used to create the output file is HOLD. The SUFFIX in the generated Master File is COMT.

TAB Format

The TAB format, introduced in Release 5.2.3, creates an output file in tab-delimited format that includes column headings in the first row. This is useful for importing data to Windows-based applications such as Microsoft Access and Excel.

The TAB format includes a built-in safety feature that allows embedded quotation marks within text fields. It inserts a second double quotation mark (") adjacent to the existing one. For example, if you input Joe "Smitty" Smith, the output will be Joe ""Smitty"" Smith. The TAB format also includes the following features:

- All trailing blanks are stripped from alpha [A*n*] fields.
- All leading blanks are stripped from numeric [/D*x.y*, /F*x.y*, /P*x.y*, and /*In*] fields.
- There is a 32K record length limit in the output file.
- A Master File is created when the command used to create the output file is HOLD. The Master File behaves exactly as in FORMAT ALPHA, except for the inclusion of double quotation marks.

TABT Format

The TABT format creates an output file in tab-delimited format that includes column headings in the first row. This is useful for importing data to Windows-based applications such as MS Access and Excel.

The TABT format includes a built-in safety feature that allows embedded quotation marks within text fields. It inserts a second double quotation mark (") adjacent to the existing one. For example, if you input Joe "Smitty" Smith, the output will be Joe ""Smitty"" Smith. The TABT format also includes the following features:

- The first row contains field names.
- All trailing blanks are stripped from alpha [A*n*] fields.
- All leading blanks are stripped from numeric [D*x.y*, F*x.y*, P*x.y*, and *In*] fields.
- There is a 32K record length limit in the output file.
- A Master File is created when the command used to create the output file is HOLD. The Master File behaves exactly as in FORMAT ALPHA, except for the inclusion of double quotation marks.

Adjusting Field Displays

In this section:

Reducing Multiple Spaces in a Field to a Single Space

Removing a Character From a Field

Removing Leading or Trailing Occurrences of a Pattern

You can use three character functions to adjust the display of fields in a report:

- SQUEEZ reduces multiple blank spaces to a single space.
- STRIP removes a character.
- TRIM removes leading or trailing occurrences of a pattern.

Functions are fully documented in the *WebFOCUS Using Functions* manual.

For a graphical implementation of these display types, see *Functions for Displaying Fields* in Chapter 2, *Developer Studio Enhancements*.

Reducing Multiple Spaces in a Field to a Single Space

How to:

Reduce Multiple Spaces to a Single Space

Example:

Reducing Multiple Spaces to a Single Space

The SQUEEZ function reduces multiple contiguous blank spaces within a character string to a single space. The resulting character string has the same length as the original string but is padded on the right with spaces.

This is useful for displaying a field that contains extra spaces, such as a virtual field that combines two fields.

Syntax: How to Reduce Multiple Spaces to a Single Space

```
SQUEEZ(length, string, outfield)
```

where:

length

Integer

Is the length in characters of *string* and *outfield*, or a field that contains the length.

string

Alphanumeric

Is the field that contains the character string, or the character string enclosed in single quotation marks.

outfield

Alphanumeric

Is the field that contains the result, or the format of the output value enclosed in single quotation marks.

In Dialogue Manager, you must specify the format. In Maintain, you must specify the name of the field.

Example: Reducing Multiple Spaces to a Single Space

The SQUEEZ function reduces multiple spaces in the NAME field to a single blank and stores the result in a field with the format A30:

```
DEFINE FILE EMPLOYEE
 NAME/A30 = FIRST_NAME | LAST_NAME;
END
TABLE FILE EMPLOYEE
 PRINT NAME AND COMPUTE
 SQNAME/A30 = SQUEEZ(30, NAME, 'A30');
 WHERE DEPARTMENT EQ 'MIS';
END
```

The output is:

NAME		SQNAME
MARY	SMITH	MARY SMITH
DIANE	JONES	DIANE JONES
JOHN	MCCOY	JOHN MCCOY
ROSEMARIE	BLACKWOOD	ROSEMARIE BLACKWOOD
MARY	GREENSPAN	MARY GREENSPAN
BARBARA	CROSS	BARBARA CROSS

Removing a Character From a Field

How to:

Remove a Character From a String

Example:

Removing Occurrences of a Character From a String (Reporting)

The STRIP function removes all occurrences of a specific character from a string. The resulting character string has the same length as the original string but is padded on the right with spaces.

This is useful for removing punctuation in a DEFINE or COMPUTE command.

Syntax: How to Remove a Character From a String

```
STRIP(length, string, char, outfield)
```

where:

length

Integer

Is the length in characters of *string* and *outfield*, or a field that contains the length.

string

Alphanumeric

Is an alphanumeric string, or the field from which the character will be removed.

char

Alphanumeric

Is the character to be removed from the string. This can be a field that contains the character, or an alphanumeric literal enclosed in single quotation marks. If it is a field, the left-most character in the field will be used as the strip character.

Note: To remove single quotation marks, use two consecutive quotation marks. You must then enclose this character combination in single quotation marks.

outfield

Alphanumeric

Is the field that contains the result, or the format of the output value enclosed in single quotation marks.

In Dialogue Manager, you must specify the format.

Example: Removing Occurrences of a Character From a String (Reporting)

The STRIP function removes all occurrences of a period (.) from the DIRECTOR field and stores the result in a field with the format A17:

```
TABLE FILE MOVIES
 PRINT DIRECTOR AND COMPUTE
 SDIR/A17 = STRIP(17, DIRECTOR, '.', 'A17');
 WHERE CATEGORY EQ 'COMEDY'
END
```

The output is:

DIRECTOR	SDIR
ZEMECKIS R.	ZEMECKIS R
ABRAHAMS J.	ABRAHAMS J
ALLEN W.	ALLEN W
HALLSTROM L.	HALLSTROM L
MARSHALL P.	MARSHALL P
BROOKS J.L.	BROOKS JL

Removing Leading or Trailing Occurrences of a Pattern

How to:

Remove Leading and Trailing Occurrences

Example:

Removing Leading Occurrences

The TRIM function removes leading and/or trailing occurrences of a pattern within a character string. This is useful for removing patterns where extracting a token may produce inconsistent results for a DEFINE or COMPUTE command.

Syntax: How to Remove Leading and Trailing Occurrences

```
TRIM(trim_where, string, string_length, pattern, pattern_length, outfield)
```

where:

trim_where

Alphanumeric

Is one of the following, which indicates where to remove the pattern:

`'L'` removes leading occurrences.

`'T'` removes trailing occurrences.

`'B'` removes both leading and trailing occurrences.

string

Alphanumeric

Is the field containing the string, or the source character string enclosed in single quotation marks.

string_length

Integer

Is the length of the string in characters.

pattern

Alphanumeric

Is the pattern to remove, enclosed in single quotation marks.

pattern_length

Integer

Is the number of characters in the pattern.

outfield

Alphanumeric

Is the field to which the result is returned, or the format of the output value enclosed in single quotation marks.

In Dialogue Manager, the format must be specified.

Example: Removing Leading Occurrences

The TRIM function removes leading occurrences of the characters BR from the DIRECTOR field and stores the result in a field with the format A17:

```
TABLE FILE MOVIES
 PRINT DIRECTOR AND
 COMPUTE TRIMDIR/A17 = TRIM('L', DIRECTOR, 17, 'BR', 2, 'A17');
 WHERE DIRECTOR CONTAINS 'BR'
END
```

The output is:

DIRECTOR	TRIMDIR
ABRAHAMS J.	ABRAHAMS J.
BROOKS R.	OOKS R.
BROOKS J.L.	OOKS J.L.

PostScript Type 1 Fonts for PS and PDF Reports

You can add and configure PostScript Type 1 fonts to significantly expand your options for displaying and printing PS and PDF reports beyond those provided by the basic set of fonts distributed with Adobe Acrobat Reader (Courier, Helvetica, and Times). Thousands of PostScript fonts are available to make your reports more stylish and useful, including some that support symbols and bar codes.

How Type 1 Fonts Are Used

WebFOCUS generates a PSD or PS document from scratch, physically embedding all the objects it displays or prints, including images and fonts, in the document itself.

When you execute a report and specify PDF or PS as your display format, the WebFOCUS Reporting Server retrieves the data and begins to format the report. Fonts and images specified in the StyleSheet must be available to the server to create the output file. It reads the font information from the font files and embeds that information into the document. The font itself is stored on the server.

To ensure that the server can locate the required information, you must define and map it in the following files:

- **Font file, usually a Printer Font Binary (PFB) file.** This file contains the information about the shape to draw for each character of the font. The information in the font file is scalable, which means that a single font file can be used to generate characters of any size. Note, however, that bold and italic variations of the typeface are separate fonts. An alternative ASCII format, PFA, can also be used by WebFOCUS.
- **Adobe Font Metrics (AFM) file.** This file is distributed with all Adobe fonts. It contains information about the size of each character in each font. WebFOCUS uses this information to lay out the report on the page. (Note that the three built-in fonts also have AFM files, which are distributed with WebFOCUS. However, these fonts do not require font files, since the fonts are built in to Acrobat.)

  Note: A Printer Metrics File (PFM) is also available. This file is used by applications such as Acrobat Reader for laying out text; however, it is not supported by WebFOCUS. You must use the AFM file.
- **WebFOCUS Font Map files.** These configuration files map the name of a font to the appropriate font and font metrics files (AFM and PFB or PFA). There are two versions of this file: PDF.FMP and PSCRIPT.FMP. You can update either one or both, depending on the output format (PDF or PS) with which you plan to use the Type 1 fonts.

Through a simple process, you can customize your environment to take advantage of these fonts.

- First, copy the font files (AFM and PFB or PFA) to a location that is accessible by the WebFOCUS Reporting Server. (The location that contains the default fonts shipped with WebFOCUS is a good choice: *drive*:/srv52/home/etc.)
- Next, update the font map files (PDF.FMP and/or PSCRIPT.FMP). These configuration files already exist in the correct location, with specifications for the fonts that are distributed with Adobe Acrobat Reader. The FMP files map the name of a font to its actual font file information. You will add the new font definitions to these files.

  Once this step is completed, you can begin using the new font in your StyleSheet declarations.

For detailed procedures and sample map files, see *Adding PostScript Type 1 Fonts for PS and PDF Reports* in the *Choosing a Display Format* chapter in the *Creating Reports With WebFOCUS Language* manual.

Controlling the Paper Source for a PS Printer

The SET PSPAGESETUP command instructs a PostScript printer to use the paper source that matches the value specified for the PAGESIZE parameter.

Syntax: How to Control the Paper Source for a PS Printer

```
SET PSPAGESETUP = {ON|OFF}
```

where:

`ON`

Includes PostScript code that automatically tells a PostScript printer how to set its paper source.

`OFF`

Does not include PostScript code for the selection of a PostScript printer paper source. OFF is the default value.

SET Parameter Summary

You can customize your WebFOCUS and Developer Studio environments with the SET command. The following SET parameters were introduced or enhanced in Release 5.2. Those marked with asterisks were introduced in Release 5.2.3.

SET commands are fully documented in the *Customizing Your Environment* chapter in the *Developing Reporting Applications* manual.

SET Parameter	Description	For details, see...
BLANKINDENT	Clarifies relationships within an FML hierarchy by indenting the captions (titles) of values at each level.	*Indent Row Titles in an FML Hierarchy* on page 1-86.
CDN	Specifies the punctuation used in numerical notation.	*Determine the Punctuation of Large Numbers* on page 1-118.
CENT-ZERO	Displays a leading zero in decimal-only numbers.	*Display a Leading Zero in Decimal-only Numbers* on page 1-120.
DEFINES	Compiles virtual fields into machine code to improve performance.	*Compile DEFINE Expressions* on page 1-106.
ERROROUT	Terminates a request and returns an error message when an error is encountered.	*Control Error Processing* on page 1-106.
EXL2KLANG	Specifies the language used on the server for Microsoft® Excel requests.	*Set the Language for Excel Requests* on page 1-137.
* FILECASE	Automatically converts file names and directories to lowercase.	*Convert File Names to Lowercase* on page 1-107.
* FOCHTMLURL	Allows you to access resources with an alias other than /ibi_html.	*Access a Resource With a Non-Standard Alias* on page 1-105.

SET Parameter	Description	For details, see...
FORMULTIPLE	Allows you to include the same FOR field value in multiple rows of an FML matrix.	*Display the Same Record in Multiple FOR Rows* on page 1-85.
GRAPHSERVURL	Creates a GIF file of the graph by sending an HTTP request to a servlet.	*Save a Graph as a GIF File* on page 1-94.
* GTREND	Specifies the use of basic linear regression to alter the X- and Y-axis values in a SCATTER graph.	*Alter X- and Y-Axis Values Using Linear Regression* on page 1-97.
HDAY	Specifies the holiday file from which to retrieve dates that are considered holidays.	*Specify Holidays* on page 1-104.
* HNODATA	Controls missing values propagated to a HOLD file.	*Control Missing Values Propagated to a HOLD File* on page 1-44.
HOLDMISS	Distinguishes between missing data and default data in a HOLD file.	*Store Missing Data in HOLD Files* on page 1-47.
JSURL	Includes JavaScript or VBScript files in an HTML report.	*Include a JavaScript or VB Script File in an HTML Report* on page 1-104.
* NULL	Enables creation of a variable-length comma or tab delimited HOLD file that differentiates between a missing value and a blank string or zero value.	*Propagate Missing Values to a Delimited HOLD File* on page 1-49.

SET Parameter	Description	For details, see...
PCOMMA	Enables the reading of PC-type comma delimited files with double quotation marks around alpha data and a carriage return line at the end of each record.	*Control Retrieval of Comma-delimited Data Sources With PCOMMA* on page 1-152.
PSPAGESETUP	Used with the PAGESIZE parameter, sets the paper size for a PDF or PostScript report.	*Control the Paper Source for a PS Printer* on page 1-161.
* SUMMARYLINES	Controls summary line processing in reports that use prefix operators.	*Control Summary Line Processing* on page 1-38.
* USERFCHK	Controls the level of verification applied to DEFINE FUNCTION and Information Builders-supplied function arguments.	*Control Function Argument Verification* on page 1-67.
* USERFNS	If your site has a locally written function with the same name as an Information Builders-supplied function, the USERFNS setting determines which function will be used.	*Enable Function Argument Verification* on page 1-67.

Functions Summary

The following functions were introduced in Release 5.2. Those marked with asterisks were introduced in Release 5.2.3.

Adjusting Field Displays

Function	Description	For details, see...
SQUEEZ	Reduces contiguous blanks to a single blank.	*Reducing Multiple Spaces in a Field to a Single Space* on page 1-155.
STRIP	Removes occurrences of a character from a string.	*Removing a Character From a Field* on page 1-156.
TRIM	Removes leading/trailing occurrences of a pattern (for example, ing).	*Removing Leading or Trailing Occurrences of a Pattern* on page 1-158.

Using Variable-Length (A*n*V) Fields With Functions

Function	Description	For details, see...
* LENV	Returns the actual length of an A*n*V input field or the size of an A*n* field.	*Find the Length of an Alphanumeric Field* on page 1-53.
* LOCASV	Converts alphabetic characters to lowercase and is similar to LOCASE.	*Create a Variable Length Lowercase String* on page 1-55.
* POSITV	Finds the starting position of a substring within a larger string and is similar to POSIT.	*Find the Beginning of a Variable Length Substring* on page 1-57.
* SUBSTV	Extracts a substring from a string and is similar to SUBSTR.	*Extract a Variable Length Substring* on page 1-59.
* TRIMV	Removes a pattern from a string and is similar to TRIM.	*Remove Characters From a String* on page 1-62.
* UPCASV	Converts alphabetic characters to uppercase and is similar to UPCASE.	*Create a Variable Length Uppercase String* on page 1-64.

Calculating Cumulative Normal Distribution

Function	Description	For details, see...
NORMSDST	Calculates the percentage of data values that are less than or equal to a normalized value.	*Calculate Cumulative Standard Normal Distribution* on page 1-23.
NORMSINV	Finds the normalized value that forms the upper boundary of a percentile in a standard normal distribution curve.	*Calculate Inverse Cumulative Standard Normal Distribution* on page 1-23.

Retrieving FOR Field Values

Function	Description	For details, see...
FMLINFO	Returns the FOR value associated with each row in an FML report. Can be used to change negative signs to positive as required by accounting data.	*Retain FOR Field Values for Consolidated Report Rows* on page 1-87.

CHAPTER 2

Developer Studio Enhancements

This chapter describes all of the features for Developer Studio in Version 5 Release 2 and has been updated to reflect Version 5 Release 2.3. Topics that contain Version 5 Release 2.3 features are marked with asterisks.

Topics:

New Architecture

* Managed Reporting Developer for Windows

Enhanced Explorer

Project Wizard

Redesigned Deploy Wizard

Dimensions Tool

Synonym Enhancements

Date Format and Time Display Options

Comma Suppression Display Option

Percent Edit Display Option

Report Viewing in a Browser

PS Format in Compound Reports

Excel Enhancements

* Comma-delimited and Tab-delimited Output Formats

Configuration for ANSI (Windows) Format

* Creating and Styling Reports With the Report Painter

Calculating Trends and Predicting Values in Developer Studio

* Designing a User Interface With the Resource Layout Painter

Source File Management

* Update Assist

Financial Report Painter Enhancements

* SET Parameters

Functions for Displaying Fields

New Architecture

Choose the Developer Studio environment that meets your development needs.

If you install a WebFOCUS Reporting Server during the Developer Studio installation procedure, you can:

- Locally develop and deploy self-service applications from the Projects area.

  Stand-alone project-based development and deployment requires installation of a WebFOCUS Reporting Server on the same machine as Developer Studio. A WebFOCUS Client is also required for project-based development. The files that you create for a local project reside in a subdirectory under APPROOT (defined in the configuration files edaserve.cfg and cgivars.wfs). The Application Root directories (APPROOT directories) attribute must point to the same directory for project-based development since files are created with the WebFOCUS Client, which resides on the Web server.

- Connect to one or more remote servers and modify existing self-service applications on those servers. For example, you can add a reporting procedure to an existing application.

- Configure access to one or more WebFOCUS environments so that you can manage resources on the WebFOCUS Client and Reporting Server, and in the Managed Reporting Repository (if installed). From the environment tree you can create and edit procedures, metadata, HTML files, and more.

If you do not install a WebFOCUS Reporting Server during the Developer Studio installation procedure, your environment allows the last two capabilities.

For differences in development environments and other architectural details, see the *Introducing WebFOCUS and Developer Studio* chapter in the *Developer Studio Application Development Getting Started* manual.

Managed Reporting Developer for Windows

Release 5.2.3 introduces the new WebFOCUS Managed Reporting Developer for Windows product. It enables Developers to create and manage domain content. Using this product, Developers can:

- Create Standard Reports and Reporting Objects with Windows-based tools such as the Report Painter.
- Analyze data with OLAP.
- Create and manage WebFOCUS Reporting Server resources such as metadata, procedures, and other files.

This product is fully documented in the *WebFOCUS Managed Reporting Developer's Manual*.

By design, all Managed Reporting administrative features are disabled. That means that Developers cannot create, delete, or rename Domains using the Managed Reporting Developer. There is also no access to User Administrator, Dashboard View Builder, or the ReportCaster Console. The Web Applications and User Management nodes have been removed in Explorer since they do not apply to this product.

Important: Managed Reporting Administrators can use this product, but their behavior is the same as that of Developers who have the Data Server privilege. Likewise, Developers still have their administrative capabilities when they use the full Developer Studio. Administrative functions are fully documented in the *WebFOCUS Managed Reporting Administrator's Manual*.

Managed Reporting Developer for Windows can be used to create self-service applications. You can create Server procedures and metadata using the Data Server node (so long as you are either an Administrator or a Developer with the Data Server privilege). But you cannot access the Web Applications folder from Explorer (which you might want to do to create launch pages) and you cannot work in the project development paradigm which includes support for source control and deployment scenarios. In these cases, you must use the full Developer Studio.

The functionality of the Managed Reporting Developer for Windows product is described in the *WebFOCUS Managed Reporting Developer's Manual*.

Enhanced Explorer

In the Explorer, you can:

- Copy and paste a procedure within a single project. The copied procedure is identified as "Copy of *procedure_name*." See *Copying a Procedure* in the *Creating a Reporting Procedure* chapter in the *Developing Reporting Applications* manual.
- Enter a long procedure name in the Add Procedure dialog box. The name is no longer restricted to eight characters. See *Selecting a Creation Tool* in the *Creating a Reporting Procedure* chapter in the *Developing Reporting Applications* manual.
- Add a virtual folder to a project, or add a virtual subfolder to an existing folder (for example, an HTML Files or Master Files folder). See *Organizing a Project* in the *Creating a Reporting Application* chapter in the *Developing Reporting Applications* manual.
- Customize the file types displayed in any folder. See *Organizing a Project* in the *Creating a Reporting Application* chapter in the *Developing Reporting Applications* manual.
- Right-click to rename a component such as a folder, subfolder, HTML file, Master File, or procedure. See *Organizing a Project* in the *Creating a Reporting Application* chapter in the *Developing Reporting Applications* manual.
- Permanently remove a file (for example, a procedure) from the hard drive by deleting it from the associated project. See *Adding a Master File to a Project* in the *Creating a Reporting Application* chapter in the *Developing Reporting Applications* manual.

In addition, the Project Properties dialog box has been expanded so that you can select a deployment scenario, customize the items displayed under a project folder, and set other project attributes. See *Viewing and Modifying Project Properties* in the *Creating a Reporting Application* chapter in the *Developing Reporting Applications* manual.

Project Wizard

The Project Wizard enables you to select a folder or folders to add to a project path, using the redesigned Browse for Folder dialog box. To use the Browse for Folder dialog box, see *Creating a Project* in the *Creating a Reporting Application* chapter in the *Developing Reporting Applications* manual.

Redesigned Deploy Wizard

Deploying a project is the process of copying project files to a Web server and a WebFOCUS Reporting Server so the application can run on the Web and be accessed by other users. HTML forms are used to launch the application in the Web environment. The Deploy Wizard guides you through the process of creating a configuration that manages the deployment of your project files to the Web. In Developer Studio, you can take advantage of the following enhancements to the Deploy Wizard:

- **Multiple deployment scenarios.** You can define multiple deployment scenarios and save them for future deployment. A deployment scenario includes the partitioning of the project files and the selection of servers. For example, you might have two deployment scenarios for a project, one that maps the files to a production server, and another that maps the files to a test server. To create multiple deployment scenarios, see *Step 2. Create a Deployment Scenario* in the *Partitioning and Deploying Project Files* chapter in the *Developing Reporting Applications* manual.
- **Consolidated deployment tool for Developer Studio and Maintain.** The Developer Studio and Maintain environments have a consolidated deployment tool. To use the deployment tool in Developer Studio, see the *Partitioning and Deploying Project Files* chapter in the *Developing Reporting Applications* manual. For details about using the deployment tool in Maintain, see the *Setting Up Your WebFOCUS Maintain Project* chapter in the *Developing WebFOCUS Maintain Applications* manual.
- **Deployment to multiple servers**. You can deploy your project files to multiple WebFOCUS Reporting or Maintain servers. This enhancement enables you to access data on multiple servers, run your report components in the most suitable environment, and speed up your application processing. It also enables greater control of access to your reporting applications. Deployment to multiple servers is fully documented in *Step 3. Partition the Project Files* in the *Partitioning and Deploying Project Files* chapter in the *Developing Reporting Applications* manual.

This feature is relevant if you develop WebFOCUS reporting applications or WebFOCUS Maintain applications and are responsible for deploying your application to end users who access it from a Web browser.

Dimensions Tool

How to:

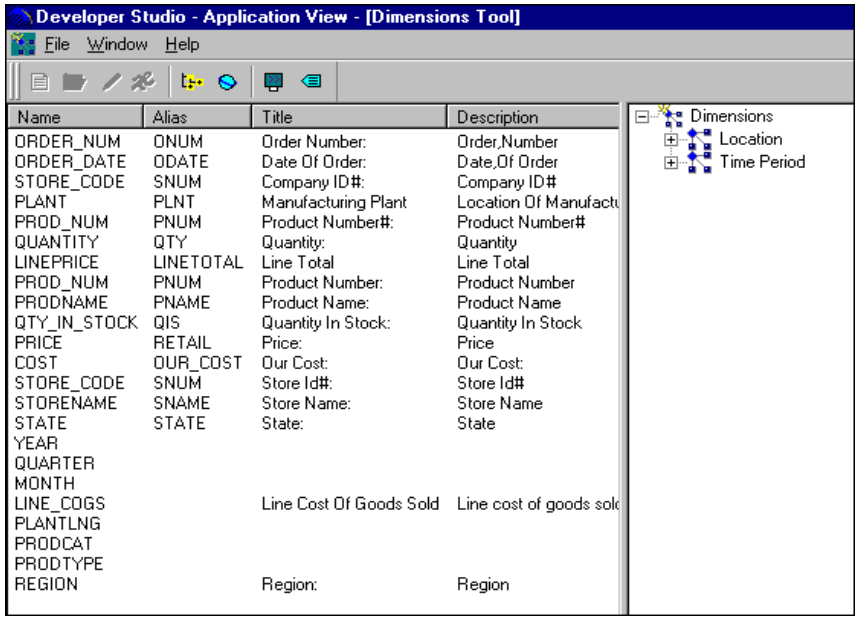
Launch the Dimensions Tool From the Procedure Window

Launch the Dimensions Tool From the Report Painter

The Dimensions Tool enables you to create OLAP hierarchies and dimensions, based on enterprise data for multi-dimensional analysis, without modifying the Master File. The new logical view is saved as part of a procedure. The Dimensions Tool works with FOCUS data sources, relational tables, and OLAP-enabled Master Files. This tool is accessed through the Dimension component, but it does not show the graphical representation of a Master File. Instead, it lists the fields. The process of creating a hierarchy is the same; drag and drop fields from the left pane to the right pane.

Procedure: How to Launch the Dimensions Tool From the Procedure Window

1. Click and hold a component connector (yellow diamond) and select *Dimensions*.
2. In the Open dialog box, select a Master File for which you want to create dimensions and click *Open*. The Dimensions Tool opens with the Master File you selected.


Developer Studio - Application View - [Dimensions Tool]
File Window Help
Name Alias Title Description
ORDER_NUM ONUM Order Number: Order,Number
ORDER_DATE ODATE Date Of Order: Date,Of Order
STORE_CODE SNUM Company ID#: Company ID#
PLANT PLNT Manufacturing Plant Location Of Manufactu
PROD_NUM PNUM Product Number#: Product Number#
QUANTITY QTY Quantity: Quantity
LINEPRICE LINETOTAL Line Total Line Total
PROD_NUM PNUM Product Number: Product Number
PRODNAME PNAME Product Name: Product Name
QTY_IN_STOCK QIS Quantity In Stock: Quantity In Stock
PRICE RETAIL Price: Price
COST OUR_COST Our Cost: Our Cost:
STORE_CODE SNUM Store Id#: Store Id#
STORENAME SNAME Store Name: Store Name
STATE STATE State: State
YEAR
QUARTER
MONTH
LINE_COGS Line Cost Of Goods Sold Line cost of goods sol
PLANTLNG
PRODCAT
PRODTYPE
REGION Region: Region
Dimensions
Location
Time Period

3. Create hierarchies the same way you do with the Dimension Builder by dragging and dropping a selected field to the appropriate level in the right pane of the Dimensions Tool window.

Procedure: How to Launch the Dimensions Tool From the Report Painter

1. From the Report Painter's Report menu, select *OLAP Dimensions*.

   The Dimensions Tool opens with the Master File you selected when you launched the Report Painter.

2. Create hierarchies the same way you do with the Dimension Builder by dragging and dropping a selected field to the appropriate level in the right pane of the Dimensions Tool window.

Synonym Enhancements

Using the Refresh Synonym feature you can recreate a synonym in order to update field information while preserving the old synonym's title, description, usage, virtual field, and DBA information. This feature also synchronizes the Master File with the table on which the synonym is based. You can access the Refresh Synonym feature from either the Projects on localhost area or the WebFOCUS Environments area in the Explorer window. This feature is also available in the Server Console.

Procedure: How to Refresh a Synonym

1. Right-click a synonym (Master File).

   - In the Projects area, Master Files are listed in a Master Files folder under a project's name.
   - In the WebFOCUS Environments folder, Master Files are listed within an application in the Data Servers Applications area, or in the Data folder under the Cataloged Path folder if the server is set up to use EDAPATH.

2. Choose one of the following:

   - *Refresh Synonym*, then *Replace* to recreate the Master File.

     or

   - *Refresh Synonym*, then *Create new…* to provide a name for the new synonym.

     When you choose this option, a new synonym is created based on the name you provide, but the original Master File or Access File is not modified.

     Synonyms for FOCUS and relational tables can have a maximum of 64 characters, but they cannot include spaces or special characters.

Date Format and Time Display Options

How to:

Change the Display of the Date Format

Change the Time Display

You can apply date format and time display options to date fields using the Date Formats dialog box, which can be accessed from the Formats dialog box. In previous releases, you could only select different date formats (for example, MDY, DMY, and so on) from the Format dialog box.

Procedure: How to Change the Display of the Date Format

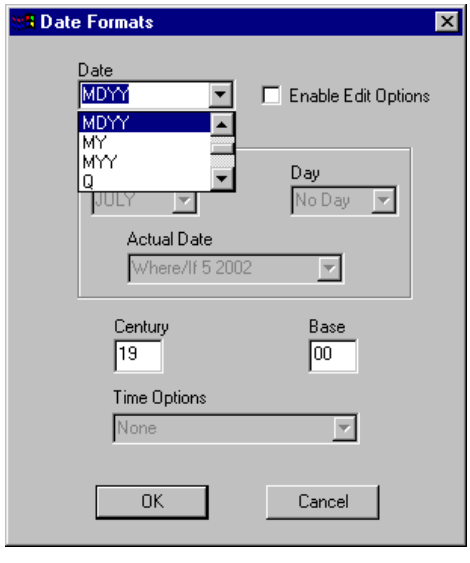
1. Select the date field that you want to change in the Report Painter window.
2. Right-click and select *Format* from the shortcut menu.

   or

   Choose *Format* from the Properties menu.

   The Format dialog box opens.
3. Click the *Date* button on the Format dialog box.

   The Date Formats dialog box opens.
4. Scroll the Date list to select a date format.



5. Click *OK*.

Procedure: How to Change the Time Display

You have the option of changing the time display when you use a date field that contains a time component in the Master File.

1. Select the date time field that you want to change in the Report Painter window.
2. Right-click and select *Format* from the shortcut menu.

   or

   Choose *Format* from the Properties menu.

   The Format dialog box opens.
3. Click the *Date* button on the Format dialog box.

   The Date Formats dialog box opens.
4. Scroll the Time Options list to select the one(s) you wish to change.
5. Click *OK*.

Comma Suppression Display Option

Reference:

Suppressing the Display of Commas in Numeric Data

You have the ability to suppress the display of commas in a numeric column using the comma suppress edit format option. This option is listed in the Format dialog box, which you can access from the Master File Editor, the Report Painter, the Define Tool, or the Compute Tool. It enables you to display numeric and monetary data without commas.

The comma suppress edit format option is available only when a numeric format, such as Decimal (D), Integer (I), Floating (F), or Packed (P), is selected in the Format Types group box.

Numeric display options are fully documented in *Numeric Formats* in the *Assigning Field Formats* chapter in the *Creating Reports With Graphical Tools* manual.

Reference: Suppressing the Display of Commas in Numeric Data

Option	Format	Data	Display
Comma suppression	D6c D7Mc D7Nc	41376 6148 6148	41376 $6148 $ 6148

Percent Edit Display Option

You can display a percent sign along with numeric data using the percent edit format option. This option is listed in the Format dialog box, which you can access from the Master File Editor, the Report Painter, the Define Tool, or the Compute Tool. It enables you to display data as percentages.

When you apply this option to a column, a percentage sign is added to the end of the specified column.

The percent edit format option is available only when the following conditions are met:

- The Decimal (D), Integer (I), Floating (F), and Packed (P) formats are selected in the Format Types group box.
- The M (floating dollar sign), N (non-floating dollar sign), or E (scientific notation) edit options are not selected in the Edit Options group box. If you select the M, N, or E options, the percent edit option becomes disabled.

Numeric display options are fully documented in *Numeric Formats* in the *Assigning Field Formats* chapter in the *Creating Reports With Graphical Tools* manual.

Reference: Displaying a Percent Sign With Numeric Data

Option	Format	Data	Display
Percent sign	I2% D7% F3.2%	21 6148 48	21% 6,148% 48.00%

Report Viewing in a Browser

Your HTML report output and graphs are displayed in your browser instead of the Desktop Viewer.

For examples of report and graph viewing options, see the *Viewing and Printing Reports and Graphs* chapter in the *Creating Reports With Graphical Tools* manual.

PS Format in Compound Reports

In addition to being able to combine multiple reports into a single PDF file, you can concatenate reports into a PS file using the SET COMPOUND command. The first PS or PDF report defines the format for the concatenated report, enabling you to intersperse intermediate reports of other formats (HTML, PDF, PS, EXL2K) into one encompassing report. You can then run or distribute the report with ReportCaster, which displays the compound PDF report in Adobe Acrobat Reader, or sends the compound PS report directly to a printer. See the ReportCaster documentation for details about this product.

For language implementation of this feature, see *Displaying Compound Reports in PS Format* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.

Excel Enhancements

The Excel 2000 format (EXL2K) has been enhanced to:

- **Display Formatted Dates and Numeric Values in Excel 2000.** Excel 2000 spreadsheets generated by WebFOCUS contain the same numeric formatting as specified in the data source's Master File or as specified in a temporary field. WebFOCUS numeric values and date formats (such as currency and Smart Dates) are translated to supported Excel formats and are displayed properly in Excel 2000.
- **Generate Native Excel Formulas in Excel 2000.** When you display or save a report request using EXL2K FORMULA, the resulting spreadsheet contains an Excel formula that computes and displays the results of any type of summed information (such as column totals, row totals, subtotals, and calculated values), rather than static numbers. Spreadsheets saved using the EXL2K FORMULA format are interactive, allowing for "what if" scenarios that immediately reflect any additions or modifications made to the data.
- **Customize Worksheet Tabs.** By default, when you choose EXL2K as your display format, the report opens in an Excel 2000 worksheet, identified in a tab at the bottom of the spreadsheet as Sheet1, Sheet2, and so on.

  You can change the name of a Sheet tab to make it more descriptive by including the TITLETEXT attribute in the StyleSheet declaration.

  Note that the same syntax changes the browser title bar in an HTML report.
- **View and Save a Report in Excel 97 Format.** The EXL97 format enables you to view and save reports in Excel 97. When specifying the EXL97 format, an HTML-based file is generated with an extension of .e97. The appropriate MIME type is automatically assigned to designate Excel as the active application for this file type.

  Format EXL97 is fully compatible with Excel 2000 and Excel 2002.

For examples and procedures on using Excel output format in Developer Studio, see *Saving Reports as Excel 2000, Excel 97, and Excel Output* in the *Saving and Reusing Report Output* chapter in the *Creating Reports With Graphical Tools* manual.

For language implementation of this feature, see *Excel Integration Enhancements* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.

Comma-delimited and Tab-delimited Output Formats

In Release 5.2.3, three additional output formats—COM, COMT, and TABT—are available in Developer Studio. They enable you to save report output to a designated directory for use with a desktop product.

- Comma-delimited data sources are sequential data sources in which field values are separated by commas. This format type can be imported into applications such as Excel or Lotus.
- Tab-delimited data sources are sequential data sources in which field values are separated by tabs.

To access these output formats in the Report Painter, choose *Output* from the Report menu. The Report Options dialog box opens at the Output tab. Click the *Hold Data*... button in the Hold/Save section. The Hold dialog box opens, where you can select a format from the Hold Format drop-down list.

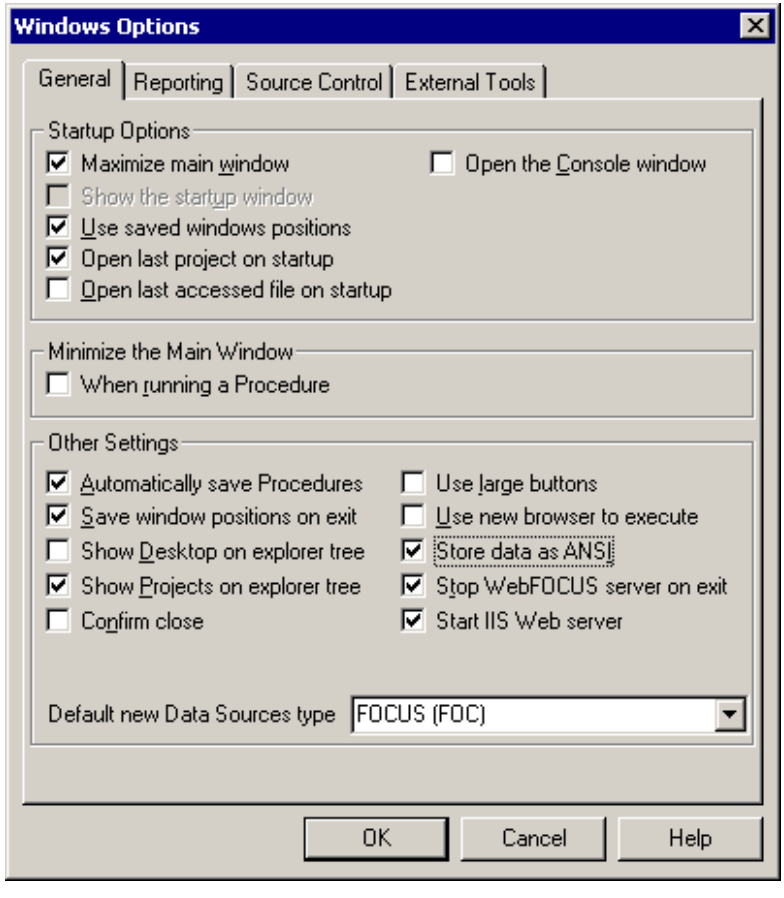
For language implementation of this feature, see *Output Formats for Application Processing* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.

Configuration for ANSI (Windows) Format

In Release 5.2.3, you can configure WebFOCUS Developer Studio to store data and all development components in ANSI (Windows) format without changing the ANSI/OEM settings in the Registry Editor.

Procedure: How to Configure Developer Studio for ANSI (Windows) Format

1. Select *WebFOCUS Developer Studio* from the WebFOCUS 52 Developer Studio program group. The Developer Studio Explorer opens.
2. Select the *Window* menu option and then choose *Options*. The Windows Options dialog box opens:



3. Select the *Store data as ANSI* check box and click *OK*.

   You need to restart WebFOCUS Developer Studio in order for this setting to take effect.

Creating and Styling Reports With the Report Painter

In this section:

Summary of Report Painter New Features

Creating Multiple Drill Downs With the Report Painter

How to:

Create a Multiple Drill Down Procedure

The Report Painter provides many powerful reporting features that enable you to create and style complex reports. You can graphically paint the report on the Report Painter window, which is a graphical representation of the report page.

The Report Painter offers an improved reporting environment, as well as greater functionality. These features are relevant if you are responsible for developing WebFOCUS reporting applications.

Summary of Report Painter New Features

The benefits of using the enhanced Report Painter include the following:

- **Added styling capabilities.** You can:
  - Drag the column border to the desired width. See *How to Increase Column Width* in the *Creating Reports With the Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.
  - Add a border to an entire report, a column, or any object area (for example, Page Heading, Page Footing, Subheading, Subfooting). You can add borders in a variety of line styles, widths, and colors. See *How to Add Borders to an HTML or PDF Report* in the *Styling Reports With the Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.

    For language implementation of this feature, see *Adding Borders for Emphasis* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.
  - Insert a spot marker. A spot marker divides text in a heading or footing into separate items. You can then individually position and style these items. See *Styling or Positioning an Individual Item in a Text String* in the *Styling Reports With the Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.

- Apply an external Cascading Style Sheet (CSS) to an HTML report. The Style tab features a Style File Selection button that enables you to apply an external Cascading Style Sheet to an HTML report. You can also assign a Cascading Style Sheet class to a report object in the StyleSheet. See *Applying an External Style Sheet* in the *Styling Reports With the Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.
- Use new shading patterns and scaling options to improve data visualization. Data visualization is supported for PDF and PS formats. Although the color option on the Data Visualization dialog box is the default for HTML, PDF, and PS formats, you can select different shading patterns for PDF and PS formats. The shading patterns make graphs in black and white reports more readable.

  There are two options for specifying relative bar graph scaling for multiple report columns under a common Across sort field to which data visualization is applied. Use the Uniform scale option if you want each vertical bar graph to be scaled based on the minimum and maximum values of all values compiled from each Across column. Use the Distinct scale option to specify that each vertical bar graph should be scaled based on the distinct minimum and maximum values for each Across column. See *Associating Bar Graphs With Columns* in the *Visualizing Trends in Reports* chapter in the *Creating Reports With Graphical Tools* manual.

- **Added formatting capabilities.** You can:
  - Style the background color for an entire report, including all column titles and all data components. You can also specify a background color for individual columns and alternating rows. See *How to Assign a Background Color to a Column* in the *Creating Reports With the Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.
  - Apply a page color. The report on the page inherits the page color. See *How to Set the Page Color for the Report* in the *Creating Reports With the Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.
  - Insert the current page number and total page count for a report as embedded fields in a report heading or footing. See *How to Add the Current Page Number* in the *Creating Reports With the Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.

    For language implementation of this feature, see *Indicating Total Number of Pages* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.

- Align decimal points. You can align decimal points when the displayed data has a varying number of decimal places. See *How to Align Decimals in a Heading or Footing* in the *Creating Reports With the Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.

  For language implementation of this feature, see *Aligning Decimals in a Multi-Line Heading* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.

- Insert the current date. You can insert the current date as an embedded field in any object area (for example, Page Heading, Page Footing, Subheading, Subfooting) in the Report Painter. Once the date is inserted, you can justify, position, and change the font of the date field. See *How to Insert the Current Date* in the *Creating Reports With the Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.

  You can also specify the date format and a display format for the time. For more information, see *Date Format and Time Display Options* on page 2-8.

- Use the Range option in the Variable Editor dialog box. You can specify a range of values instead of a list of acceptable values when you access the Variable Editor dialog box. See *How to Specify a Range of Values for Variables in Your Report* in the *Creating Reports With the Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.

- **Added dialog box options.** You can:

  - Select a column component (Title, Data, or Title and Data) and apply styling options (font and font color, grid, border, or background color), using the new Style tab on the Field Properties dialog box. In addition, you can create a condition and apply to it any style available on the Style tab. For Style tab options, see *Field Properties Style Tab* in the *Report Painter Basics* chapter in the *Creating Reports With Graphical Tools* manual. For procedures, see *Styling With Fonts, Colors, and Grids* in the *Styling Reports With the Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.

  - Copy an existing drill-down component to a column component, using the new Drill Down tab on the Field Properties dialog box. You can also open a child report from this tab for viewing or modification in a new instance of Report Painter. For Drill Down tab options, see *Field Properties Drill Down Tab* in the *Report Painter Basics* chapter in the *Creating Reports With Graphical Tools* manual. For procedures, see *Creating a Drill Down Procedure* in the *Creating Reports With the Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.

  - Remove an underline from a column title on a report, using the new General tab on the Field Properties dialog box. See *Field Properties General Tab* in the *Report Painter Basics* chapter in the *Creating Reports With Graphical Tools* manual.

- Enter replacement text using the Output tab. The Output tab on the Report Options dialog box features a WebFOCUS Title input box. The text you enter into this input box replaces the default text in the Internet Explorer title bar when you run the report in HTML format. For details, see *Report Options Output Tab* in the *Report Painter Basics* chapter in the *Creating Reports With Graphical Tools* manual.

  If you run the report in the EXL2K format, the text in the WebFOCUS Title input box replaces the default Worksheet tab text in Excel 2000.

- Use the Display Format option. The Output tab on the Report Options dialog box provides a Display Format drop-down list that enables you to specify output formats such as HTML, PDF, Excel 2000, Excel 2000 PivotTable, Excel 2000 Formula, Excel 97, User, and FOCUS default. See *Report Options Output Tab* in the *Report Painter Basics* chapter in the *Creating Reports With Graphical Tools* manual.

- Enable Excel 2000 PivotTable format from the Object Inspector. When you select Excel 2000 PivotTable in the Output tab's Display Format drop-down list, a Pivot tab is added to the Object Inspector. You can then drag available fields into dimensions within the Object Inspector's Pivot hierarchy. For Output tab options, see *Report Options Output Tab* in the *Report Painter Basics* chapter in the *Creating Reports With Graphical Tools* manual. For an example of enabling this format from the Pivot tab, see *How to Populate a Pivot Table* in the *Saving and Reusing Report Output* chapter in the *Creating Report With Graphical Tools* manual.

- **Improved WYSIWYG environment.** The Report Painter window displays the actual position of columns that have been wrapped, truncated, or set to maximum or minimum column width. For procedures on formatting these columns with these features, see *How to Increase Column Width, How to Wrap Data Automatically by Changing a Column's Width*, and *How to Truncate Report Column Values* in the *Creating Reports With the Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.

- **Enhanced Graphical User Interface (GUI).** You can align embedded fields in object areas (Page Heading, Page Footing, Subheading, Subfooting) with report columns.

  Note: This feature is available only for HTML reports.

  You can:

  - Copy style characteristics from one column to other columns using the Quick Style toolbar. You can copy font, grid, background color, conditional styling, or all of these characteristics. For procedures on copying style characteristics to other columns, see *Formatting a Column* in the *Creating Reports With the Report Painter* chapter in the *Creating Reports With Graphical Tools* manual. For procedures on copying conditions, see *Defining a Conditional Report Style* in the *Styling a Report With the Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.
  - Launch procedure components from the Prefix tab in the Object Inspector. The Prefix tab lists the components that precede the report component. Click the component to access the appropriate tool (Define, Join, or Dimension). See *Opening Report Components (Object Inspector)* in the *Report Painter Basics* chapter in the *Creating Reports With Graphical Tools* manual.
  - View the Master File structure (segments or fields) from the Fields tab in the Object Inspector. You can drag fields from this tab to the Report Painter window. If you drag a segment, all the fields in the selected segment are added to the report. See *Viewing Master File Segments (Object Inspector)* in the *Report Painter Basics* chapter in the *Creating Reports With Graphical Tools* manual.
  - View all the parts of the expression as you build it. With the new Expression Builder, drag and drop the field in the expression and select the logical relation and comparison type from drop-down lists. For procedures, see *Creating a List of Acceptable Values* in the *Creating Reports With the Report Painter* chapter in the *Creating Reports With Graphical Tools* manual. References, dialog boxes, and other procedures are fully documented in *Using the Expression Builder* in the *Writing Expressions* chapter in the *Creating Reports With Graphical Tools* manual.

- **Improved handling of images.** The Report Painter:

  - Supports layering for the display of images with other report components.
  - Tiles a background image instead of enlarging the image to fit the background.

- **General functionality improvements.** The Report Painter:
  - Enables you to save a report from the Save button on the General toolbar. The Save button saves all the components in the procedure, not just the report component.
  - Enables a developer to assign a variable as the display format. This feature enables a user to select the report's output format. See *Choosing an Output Format* in the *Creating Reports With the Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.
- **Calculated trends and predicted values.** You can calculate trends in data and predict values beyond the range of values stored in the data source with the Forecast feature. The Forecast feature uses averages, or a linear regression line, to distinguish trends and predict values. This is useful for predicting values that may occur beyond the current data set. See *Calculating Trends and Predicting Values With FORECAST in Developer Studio* on page 2-26.

  For language implementation of this feature, see *Calculating Trends and Predicting Values With FORECAST* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.
- **Apportioned numeric data in tiles.** You can group numeric data into any number of tiles (percentiles, quartiles, deciles, and so on) in tabular reports. For example, you can group student test scores into deciles to determine which students are in the top ten percent of the class.

  Grouping is based on the values in the selected vertical (BY) sort field and is apportioned as equally as possible into the number of tile groups you specify.

  If you want to group data into tiles from the Report Painter, see *How to Group Numeric Data Into Tiles* in the *Creating Reports With the Report Painter* chapter in the *Creating Reports With Graphical Tools* manual. If you want to do this from an OLAP report, see *How to Group Data Into Tiles in an OLAP Report* in the *Analyzing Data in an OLAP Report* chapter in the *Creating Reports With Graphical Tools* manual.

  For language implementation of this feature, see *Grouping Numeric Data Into Tiles* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.
- **Apportioned numeric data in ranges.** In Release 5.2.3, when you sort a report using a numeric sort field, you can group the sort field values together and define groups of equal range using the In-Ranges-Of option in the Report Painter. See *How to Group Numeric Data Into Ranges* in the *Creating Reports With the Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.

  For language implementation of this feature, see *Grouping Sort Field Values in Ranges* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.

- **Navigation of sort groups from a table of contents.** You can add multiple BY fields to an HTML Table of Contents (TOC). In the previous release, you could only sort on the highest level BY field in a single request. With the implementation of this multi-level feature, the TOC option is available when you right-click any BY field in your report.

  For this feature to be useful, the report must contain at least one vertical sort (BY) field. If you include more than one sort field in a report, the hierarchy is determined by the order in which the fields are specified in the request. The TOC displays, as hyperlinks, all values of the first (highest level) vertical sort field, as well as the values of any lower level BY fields that you designate for inclusion. Unless otherwise specified in the request, a new page begins when the highest level sort field changes.

  The TOC itself is an object that appears as an icon in the upper left corner of the report, or as one or more drop-down lists in a heading or footing.

  For examples and additional information on the multi-level HTML Table of Contents (TOC) feature, see *Navigating Sort Groups From a Table of Contents* in the *Creating Reports With the Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.

  For language implementation of this feature, see *Navigating Reports From a Table of Contents* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.

- **Check button to run procedures against the default server.** The Check button replaces the Run button that appeared in the Define, Set, Use, Filedef, and Let tools. When you click *Check*, the current procedure is run against the default server. A new dialog box opens. It displays the component's code, and either an error message or text stating that no error exists.

- **Increased Number of Columns Generated by Across.** In Release 5.2.3, each Across selection can generate up to 1,056 columns of data. The total number of Across columns is equal to the total number of horizontal sort field values multiplied by the total number of display fields.

Creating Multiple Drill Downs With the Report Painter

In Release 5.2.3, you can create multiple hyperlinks for a single value on a summary report, chosen from any combination of supported actions. For example, you can create hyperlinks that run a detail report or Maintain procedure, execute a JavaScript function, and link to a URL. When you click a field in the summary report, a pop-up menu appears, with the possible drill down options.

The multiple drill down feature applies to:

- HTML reports and format HTMTABLE. In the Output tab on the Report Options dialog box, the Select Format option must be HTML.
- Data fields in the body of the report, not to headings, subheadings, footings, or subfootings.

For language implementation of this feature, see *Specifying Multiple Drill-Down Links* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.

Procedure: How to Create a Multiple Drill Down Procedure

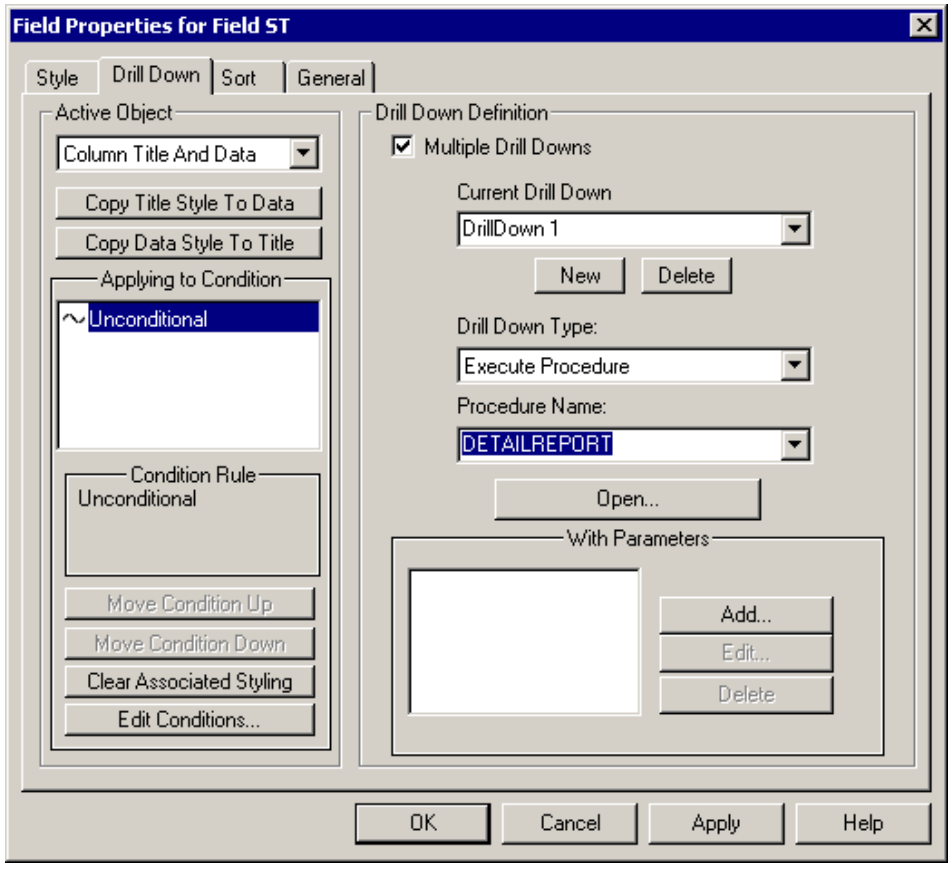
1. Create a new procedure or open an existing procedure in the Report Painter window.
2. Right-click the report component for which you will drill down.
3. Select *Options* from the shortcut menu.

   The Properties dialog box opens.
4. Click the *Drill Down* tab.

5. Select the *Multiple Drill Downs* check box.

The Current Drill Down input box is enabled and the default drill down name, DrillDown1, appears.

Tip: The pop-up menu of drill down options is only visible in the report output if the Multiple Drill Downs check box is enabled and you have at least one drill down definition.



6. Type the first drill down name in the Current Drill Down input box.

This is the name that appears on the pop-up menu in the report output.

7. Select a drill down type from the Drill Down Type drop-down list.

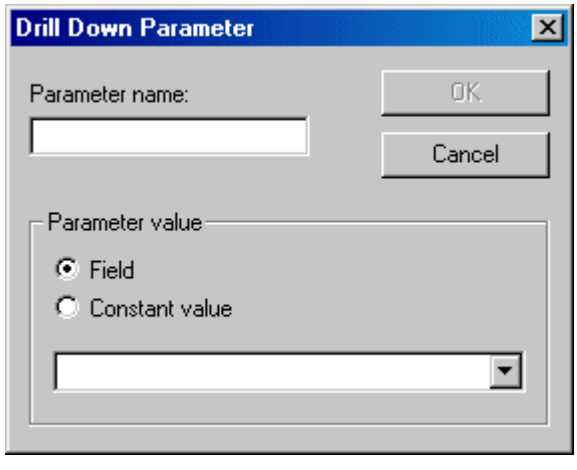
   The drill down type can be any of the following:

   - Executable procedure.
   - URL.
   - URL from a field.
   - JavaScript function.
   - Maintain procedure.
   - Compiled Maintain procedure.
   - Maintain case.

8. Enter any values required by the selected drill down type. For example, if the drill down type is Execute Procedure, the Procedure Name drop-down list appears; select the name of the detail procedure.

9. Click *Add* if you need to pass a parameter from the main report.

   The Drill Down Parameter dialog box opens.



   For this sample procedure, click *Cancel* to return to the Properties dialog box.

10. After creating the first drill down definition, click *New* on the Properties dialog box.

    When you click *New*, the default name for the next drill down, DrillDown2, appears in the Current Drill Down input box, then DrillDown3, and so on. If you change the default name of the first drill down definition, DrillDown 1, that value appears again when you create the second drill down definition.

11. Type an additional drill down name in the Current Drill Down input box.

When you create multiple drill down items, the Multiple Drill Downs check box is automatically checked and disabled to avoid the inadvertent loss of drill down items you previously created. If you delete drill down items until one remains, the Multiple Drill Downs check box is enabled again.

12. Select a drill down type from the Drill Down Type drop-down list.

13. Enter any values or parameters required by the selected drill down type.

14. To continue adding drill down items, click *New* and repeat steps 6-9.

15. When you are finished, click *OK* to return to the Report Painter window.

16. Save and run the report by clicking the *Save* and *Run* buttons on the toolbar.

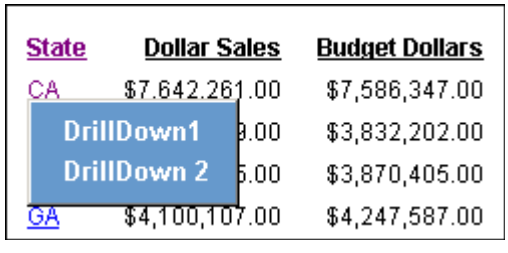
The main report appears in the browser.

17. To view the pop-up menu of drill down options:

a. Place the cursor over one of the highlighted components to reveal the drill down icon (a hand).

b. Click the component.

The pop-up menu appears.



18. Click each of the menu options to open the corresponding drill down item.

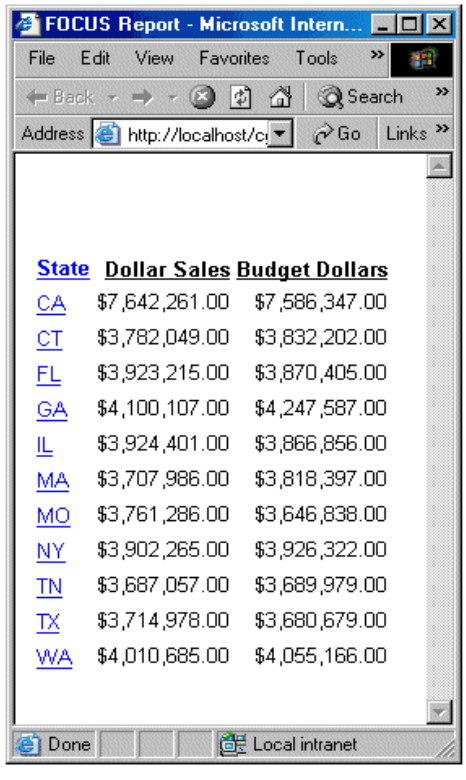
The currently selected option appears in red and is underlined. An additional browser window opens with your output.

19. Click *Back* on your browser to return to the main report.

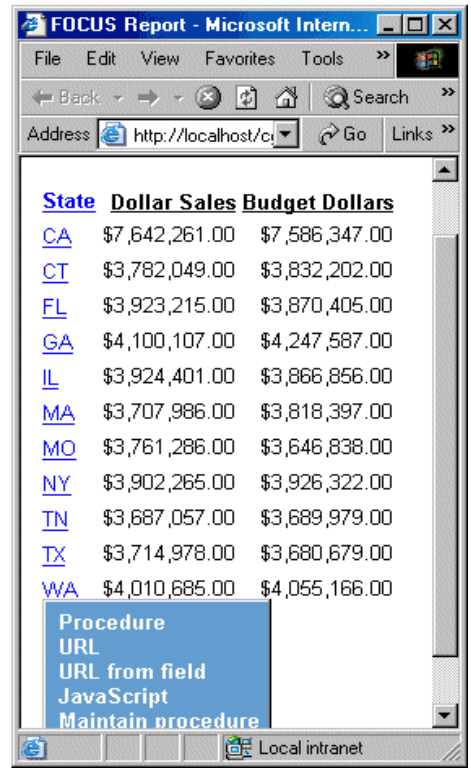
20. Click a highlighted component to redisplay the pop-up menu.

If you click the last field of the report output, scroll bars appear in the browser window if they are required for access to all of the menu options:

Before



After



For examples and details on implementing this feature, see *Creating a Multiple Drill Down Procedure* in the *Creating Reports With the Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.

Calculating Trends and Predicting Values in Developer Studio

You can calculate trends in data, and predict values beyond the range of those stored in the data source, with the Forecast and Regress commands.

- **Forecast** uses either averages or a linear regression line to distinguish trends in data, using a single independent variable to derive the dependent variable. This feature is useful for predicting values that may occur beyond the current data set.
- **Regress** uses up to three independent variables to derive a linear equation that best fits a set of numeric data points. This feature is useful if the dependent variable can be approximated by a linear combination of multiple independent variables.

For language implementation of this feature, see *Calculating Trends and Predicting Values* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.

Calculating Trends and Predicting Values With FORECAST in Developer Studio

How to:

Calculate a Simple Moving Average Column

Calculate a Single Exponential Smoothing Column

Calculate a Double Exponential Smoothing Column

Calculate a Triple Exponential Smoothing Column

Calculate a Linear Regression Column

Reference:

Forecast Dialog Box - Moving Average

Forecast Dialog Box - Exponential Average

Forecast Dialog Box - Double Exponential Average

Forecast Dialog Box - Triple Exponential Average

Forecast Dialog Box - Linear Regression

The calculations you can make to identify trends and forecast values are:

- **Simple moving average** calculates a series of arithmetic means using a specified number of values from a field.

- **Exponential moving average** calculates a weighted average between the previously calculated value of the average and the next data point. There are three methods for using an exponential moving average:
  - **Single exponential smoothing** calculates an average that enables you to choose weights to apply to newer and older values.
  - **Double exponential smoothing** accounts for the tendency of data to either increase or decrease over time without repeating.
  - **Triple exponential smoothing** accounts for the tendency of data to repeat itself in intervals over time.
- **Linear regression analysis** derives the coefficients of a straight line that best fits the data points and uses this linear equation to estimate values.

When predicting values in addition to calculating trends, the Forecast command continues the same calculations beyond the data points by using the generated trend values as new data points. For the linear regression technique, the calculated regression equation is used to derive trend and predicted values.

Forecast performs the calculations based on the data provided, but decisions about their use and reliability are the user's responsibility. Therefore, Forecast predictions are not always reliable, and many factors determine how accurate a prediction will be.

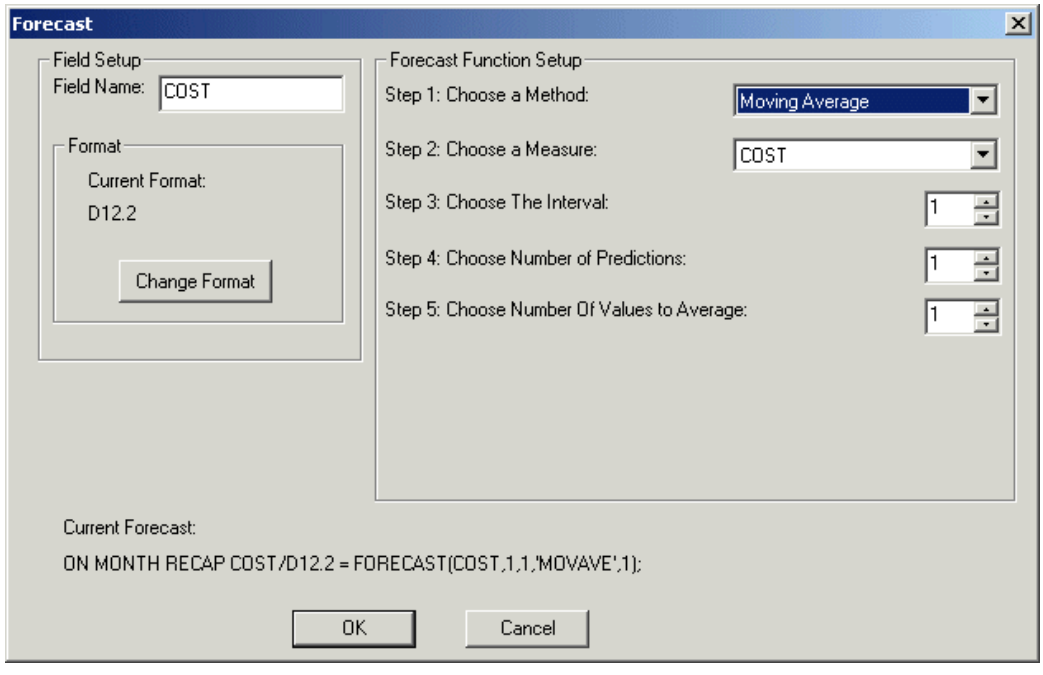
Procedure: How to Calculate a Simple Moving Average Column

1. With the By or Across field you want to use for your calculations, click *Forecast*.

   The Forecast dialog box opens.
2. If you want to change the name of the output field from FRCSTFLD, enter a new name in the Field Name field.
3. Select *Moving Average* from the Step 1: Choose a Method drop-down list.
4. Select an input field from the Step 2: Choose a Measure drop-down list.

   If you select the same field as the By or Across field, this field is not displayed even if it is included in a display command.
5. Select the increment to count each instance of the By or Across field from the Step 3: Choose The Interval spin box.
6. Select the number of predictions to be calculated from the Forecast field from the Step 4: Choose Number of Predictions spin box.
7. Select the number of values to average from the Step 5: Choose Number of Values to Average spin box.
8. Optionally, change the default field format by clicking *Change Format* and selecting the format from the Format dialog box.
9. Click *OK*.

Reference: Forecast Dialog Box - Moving Average



Forecast dialog box options are fully documented in *Calculating Trends and Predicting Values With Forecast* in the *Creating Temporary Fields* chapter in the *Creating Reports With Graphical Tools* manual.

Procedure: How to Calculate a Single Exponential Smoothing Column

1. With the By or Across field you want to use for your calculations, click *Forecast*.

   The Forecast dialog box opens.

2. If you want to change the name of the output field from FRCSTFLD, enter a new name in the Field Name box.

3. Select *Exponential Average* from the Step 1: Choose a Method drop-down list.

4. Select an input field from the Step 2: Choose a Measure drop-down list.

   If you select the same field as the By or Across field, this field is not displayed even if it is included in a display command.

5. Select the increment to count each instance of the By or Across field from the Step 3: Choose The Interval spin box.

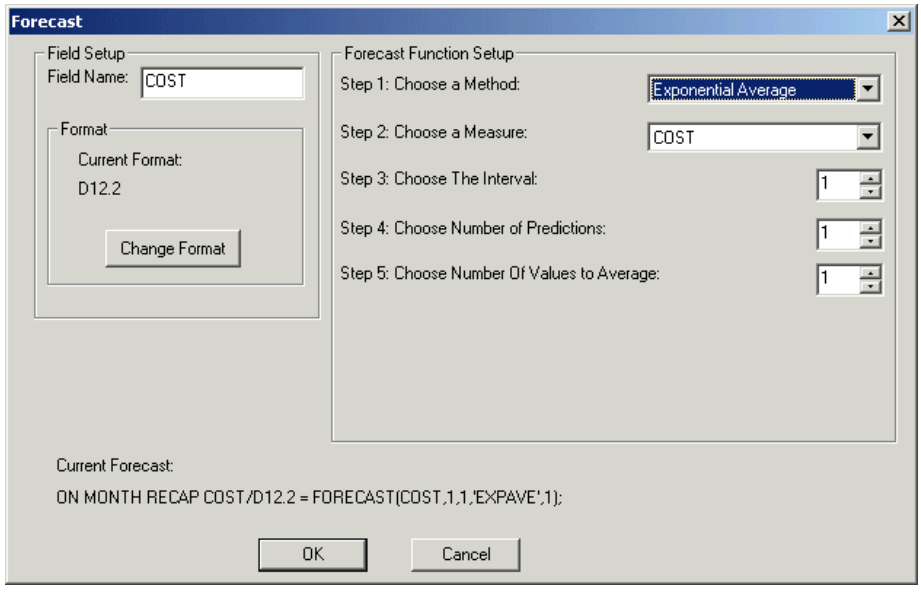
6. Select the number of predictions to be calculated from the Forecast field from the Step 4: Choose Number of Predictions spin box.

7. Select the number of values to average from the Step 5: Choose Number of Values to Average spin box.

8. Optionally, change the default field format by clicking *Change Format* and selecting the format from the Format dialog box.

9. Click *OK*.

Reference: Forecast Dialog Box - Exponential Average



Forecast dialog box options are fully documented in *Calculating Trends and Predicting Values With Forecast* in the *Creating Temporary Fields* chapter in the *Creating Reports With Graphical Tools* manual.

Procedure: How to Calculate a Double Exponential Smoothing Column

1. With the By or Across field you want to use for your calculations, click *Forecast*.

   The Forecast dialog box opens.

2. If you want to change the name of the output field from FRCSTFLD, enter a new name in the Field Name box.

3. Select *Double Exponential Average* from the Step 1: Choose a Method drop-down list.

4. Select an input field from the Step 2: Choose a Measure drop-down list.

   If you select the same field as the By or Across field, this field is not displayed even if it is included in a display command.

5. Select the increment to count each instance of the By or Across field from the Step 3: Choose The Interval spin box.

6. Select the number of predictions to be calculated from the Forecast field from the Step 4: Choose Number of Predictions spin box.

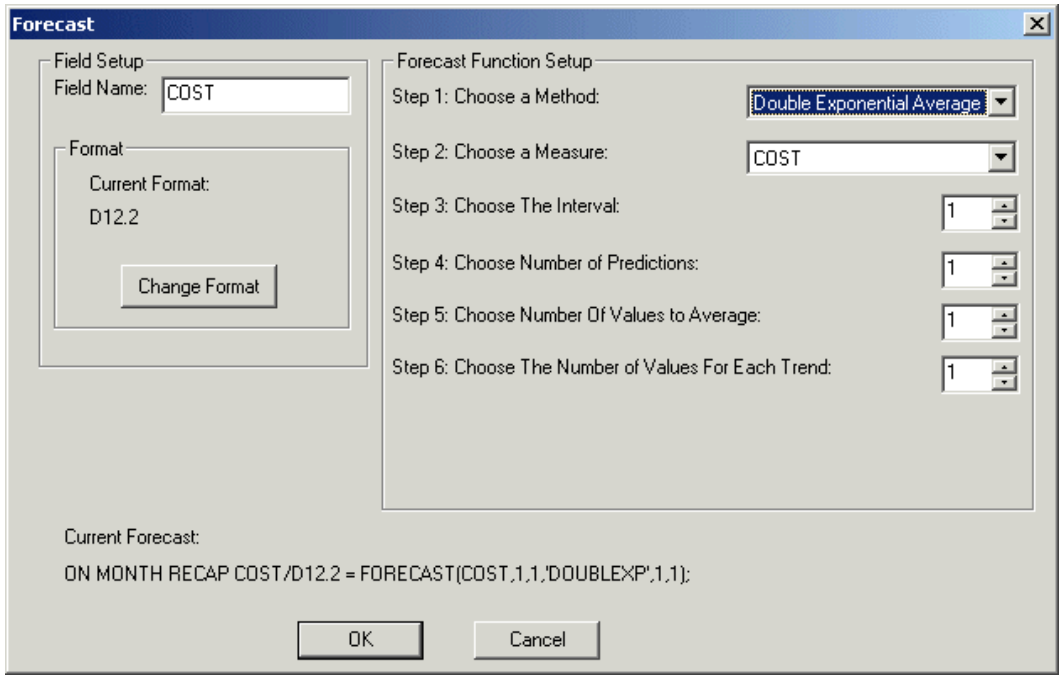
7. Select the number of values to average from the Step 5: Choose Number of Values to Average spin box.

8. Select the number to use to calculate the weights for each term in the trend from the Step 6: Choose The Number of Values For Each Trend spin box.

9. Optionally, change the default field format by clicking *Change Format* and selecting the format from the Format dialog box.

10. Click *OK*.

Reference: Forecast Dialog Box - Double Exponential Average



Forecast dialog box options are fully documented in *Calculating Trends and Predicting Values With Forecast* in the *Creating Temporary Fields* chapter in the *Creating Reports With Graphical Tools* manual.

Procedure: How to Calculate a Triple Exponential Smoothing Column

1. With the By or Across field you want to use for your calculations, click *Forecast*.

   The Forecast dialog box opens.

2. If you want to change the name of the output field from FRCSTFLD, enter a new name in the Field Name box.

3. Select *Triple Exponential Average* from the Step 1: Choose a Method drop-down list.

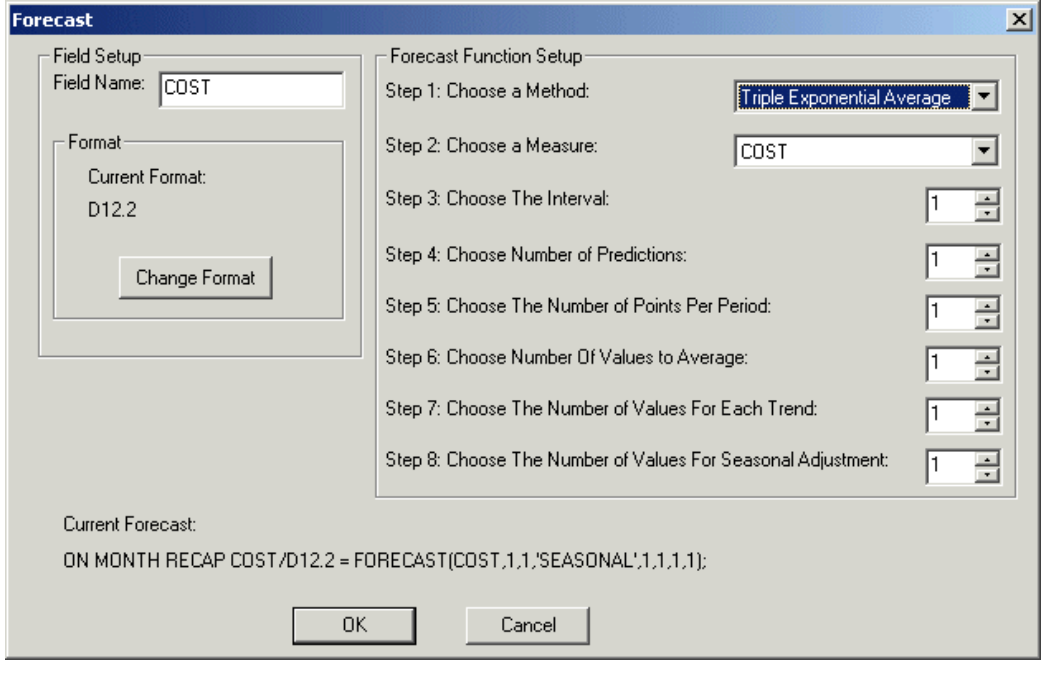
4. Select an input field from the Step 2: Choose a Measure drop-down list.

   If you select the same field as the By or Across field, this field is not displayed even if it is included in a display command.

5. Select the increment to count each instance of the By or Across field from the Step 3: Choose the Interval spin box.

6. Select the number of predictions to be calculated from the Forecast field from the Step 4: Choose Number of Predictions spin box.
7. Select the number of data points for a period from the Step 5: Choose The Number of Points Per Period spin box.
8. Select the number of values to average from the Step 6: Choose Number of Values to Average spin box.
9. Select the number to use to calculate the weights for each term in the trend from the Step 7: Choose The Number of Values For Each Trend spin box.
10. Select the number to use to calculate the weights for each term in the trend from the Step 8: Choose The Number of Values For Seasonal Adjustment spin box.
11. Optionally, change the default field format by clicking *Change Format* and selecting the format from the Format dialog box.
12. Click *OK*.

Reference: Forecast Dialog Box - Triple Exponential Average



Forecast dialog box options are fully documented in *Calculating Trends and Predicting Values With Forecast* in the *Creating Temporary Fields* chapter in the *Creating Reports With Graphical Tools* manual.

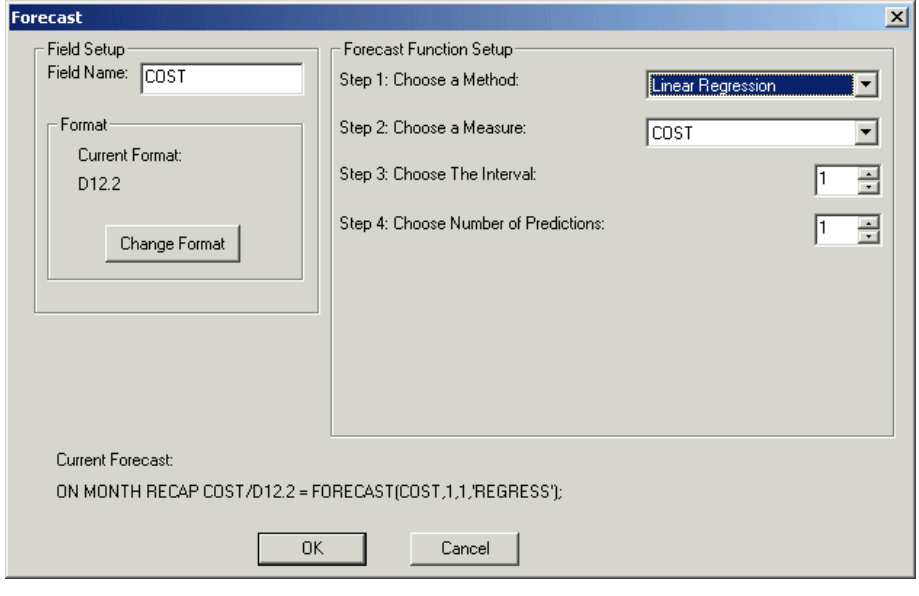
Procedure: How to Calculate a Linear Regression Column

1. With the By or Across field you want to use for your calculations, click *Forecast*.

   The Forecast dialog box opens.
2. If you want to change the name of the output field from FRCSTFLD, enter a new name in the Field Name box.
3. Select *Moving Average* from the Step 1: Choose a Method drop-down list.
4. Select an input field from the Step 2: Choose a Measure drop-down list.

   If you select the same field as the By or Across field, this field is not displayed even if it is included in a display command.
5. Select the increment to count each instance of the By or Across field from the Step 3: Choose The Interval spin box.
6. Select the number of predictions to be calculated from the Forecast field from the Step 4: Choose Number of Predictions spin box.
7. Optionally, change the default field format by clicking *Change Format* and selecting the format from the Format dialog box.
8. Click *OK*.

Reference: Forecast Dialog Box - Linear Regression



Forecast dialog box options are fully documented in *Calculating Trends and Predicting Values With Forecast* in the *Creating Temporary Fields* chapter in the *Creating Reports With Graphical Tools* manual.

Designing a User Interface With the Resource Layout Painter

In this section:

Form Controls With User-supplied Values

Enhanced Customization of Web Objects

Setting Static Parameter Values

Updating Parameters in the Resource Layout Painter With Report Painter Changes

Automatically Deleting a Form Control Associated With a Parameter

Filtering Dynamic Parameter Values

Adding an Image With a Hyperlink to the Layout

Displaying Radio Buttons and Check Box Lists in Multiple Columns

Specifying Different Names for Dynamic Parameter Values

Hiding Borders Around Frames

Applying Parameter Values to External Reports

The Resource Layout Painter, previously the Layout tool, enables you to graphically create an HTML page that incorporates WebFOCUS forms, reports, graphs, and other Web objects. The Resource Layout Painter enables you to add these components to your HTML page in an integrated process within Developer Studio.

The following topics describe the improved functionality of the Resource Layout Painter.

Form Controls With User-supplied Values

Add WebFOCUS form controls such as radio buttons, text boxes, and single and multiple drop-down lists that enable the user to supply variable values to a report or graph request. The Resource Layout Painter creates these WebFOCUS form controls automatically when you add a variable to a request in the Report Painter. For procedures on adding form controls, see *Using Forms to Supply Parameter Values* in the *Designing a User Interface for a Web Application With the Resource Layout Painter* chapter in the *Developing Reporting Applications* manual.

Enhanced Customization of Web Objects

Add more Web objects and customize the objects on your HTML page more easily and precisely. For procedures on customizing Web objects, see *Customizing the Layout* in the *Designing a User Interface for a Web Application With the Resource Layout Painter* chapter in the *Developing Reporting Applications* manual.

Setting Static Parameter Values

When you specify a list of static values for a parameter, you can select the value to pass to the parameter, the text that will represent that value in the form, and the default value to be selected.

Procedure: How to Specify Static Parameter Values

In the Parameters window:

1. In the Control Values section, select *Static*.
2. Click the *New* icon, and enter the parameter value information in the Static values section:
   a. In the Value column, enter the value to be passed to the selected parameter.
   b. In the Display column, enter the text that will represent the parameter value in the form the user views.
   c. In the Selected column, check the box for the value you want to be selected by default. If the form is a multi-select form, more than one value can be selected.
3. Repeat Step 2 until the list contains all values you want to include.
4. Optionally, click *Delete* to eliminate any values, and the up and down arrows to rearrange the order of the values.

Updating Parameters in the Resource Layout Painter With Report Painter Changes

How to:

Determine Whether to Update Changes Made With the Report Painter

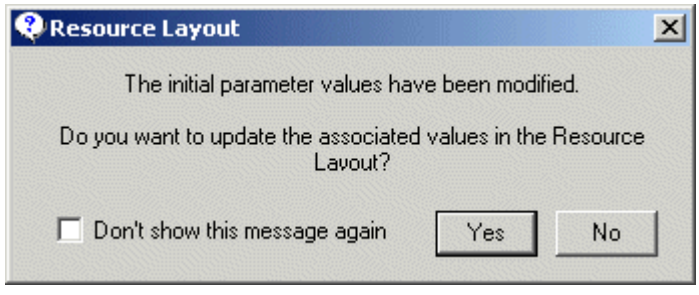
Automatically Update Parameters in the Resource Layout Painter With Report Painter Changes

When using the Resource Layout Painter, if changes are made to parameter values in the Report Painter, you have the option to update the values in the Resource Layout Painter with those entered in the Report Painter. You can also select whether to automatically update parameter values in the Resource Layout Painter.

Procedure: How to Determine Whether to Update Changes Made With the Report Painter

1. After making changes, close the Report Painter to return to the Resource Layout Painter.

   A dialog box opens, prompting you to decide whether to override parameter values in the Resource Layout Painter with the values changed in the Report Painter:

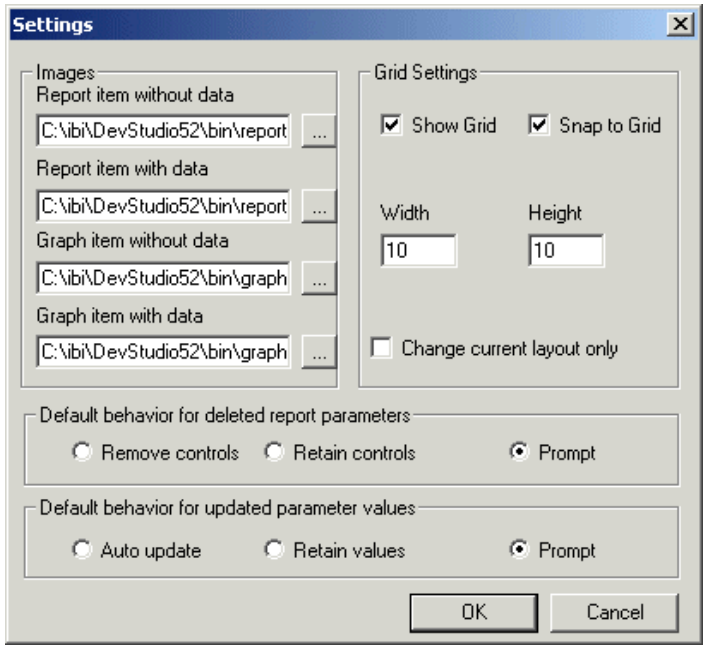

Resource Layout
The initial parameter values have been modified.
Do you want to update the associated values in the Resource Lavout?
Don't show this message again
Yes
No

2. Select *Yes* to use the values entered in Report Painter, or select *No* to ignore those changes.
3. Optionally, select *Don't show this message again* to make your selection valid for every instance in which Report Painter changes the parameter values in Resource Layout Painter.

Procedure: How to Automatically Update Parameters in the Resource Layout Painter With Report Painter Changes

1. Select *Settings...* from the Tools menu.

The Settings dialog box opens:



2. In the Default behavior for updated parameter values, select one of the following:

Auto update replaces the values in Resource Layout Painter with the modified values from Report Painter.

Retain values does not replace the values in Resource Layout Painter with the modified values from Report Painter. The original values are retained.

Prompt prompts you for a choice every time changes in the Report Painter may affect the values in Resource Layout Painter.

You can change this setting at any time.

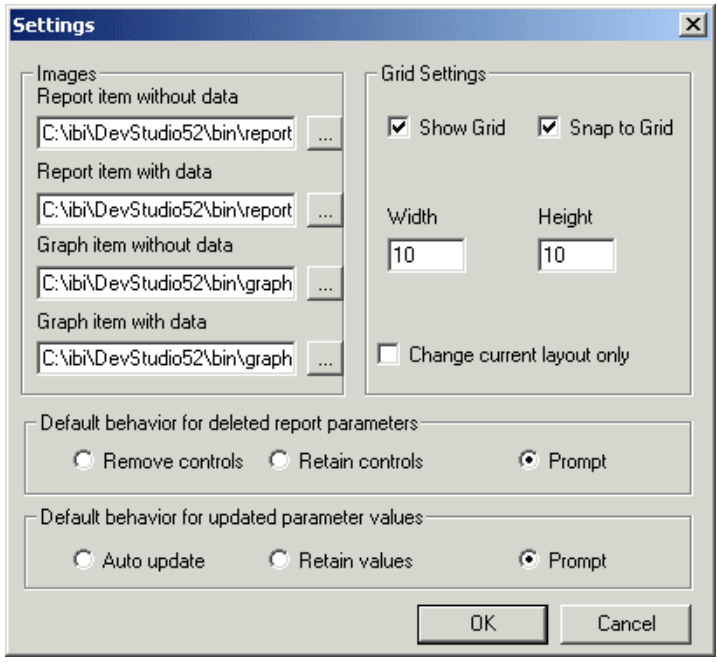
Automatically Deleting a Form Control Associated With a Parameter

When you delete a parameter from your report, you can elect to automatically delete the form control associated with that parameter. This functionality is not new, but the way this choice is made has changed.

Procedure: How to Automatically Delete a Form Control Associated With a Parameter

1. Select *Settings...* from the Tools menu.

   The Settings dialog box opens:



2. In the Default behavior for deleted report parameters, select one of the following:

   - *Remove controls* automatically deletes the form control associated with the deleted parameter.
   - *Retain controls* does not delete the form control associated with the deleted parameter.
   - *Prompt* prompts you for a choice every time a parameter is deleted.

   or

   When exiting the Report Painter after a parameter has been deleted, you are prompted to decide whether or not to delete the associated form control. Select *Don't show this message again*, then select *Yes* or *No*.

Filtering Dynamic Parameter Values

How to:

Filter Parameter Values

Reverse the Order of Chained Parameter Values

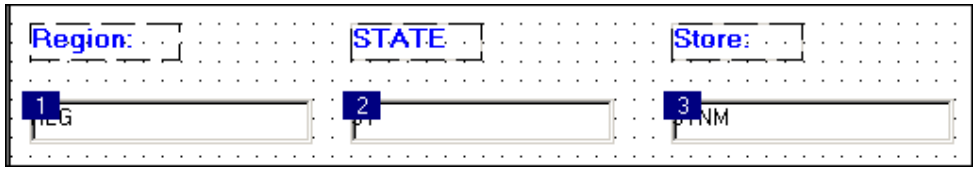
Remove a Parameter From the Chain

Cache Parameter Data

Release 5.2.3 introduces the Resource Layout Painter's chain feature to associate two or more related parameters. Chained parameter values are filtered as selections are made to each parameter control. For example, if you chain the PLANT parameter to the STATE parameter, only PLANT values for the currently selected STATE parameter are available instead of all the plants in the data source. Each time a selection is made, all chained parameters are dynamically updated. The chain feature also enables you to add, remove, and reverse the order of parameters in the chain.

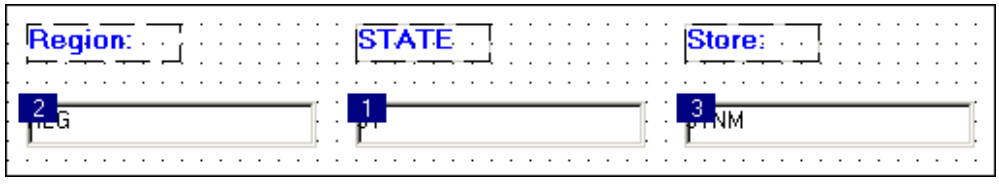
Procedure: How to Filter Parameter Values

1. Open the Resource Layout Painter.
2. In the Control Values section of the Parameters window, select the *Dynamic* radio button.
3. Press the Shift key and then click two or more parameters to select them simultaneously and activate the chain feature buttons on the layout toolbar.
4. Click the *Add to chain* button. The parameters you selected are labeled with numbers to indicate the order in which they are chained when you run your report.



Procedure: How to Reverse the Order of Chained Parameter Values

1. Click the *Chain order* button in the layout toolbar.
2. Click the parameter you want to display first in the chain.



You can continue to reverse the order in the chain by clicking the numeric labels on the chained parameters. If you would like to see the numeric labels on the chained parameters, select the *Chain order* button. Double-click a parameter to designate it as the first parameter in the chain.

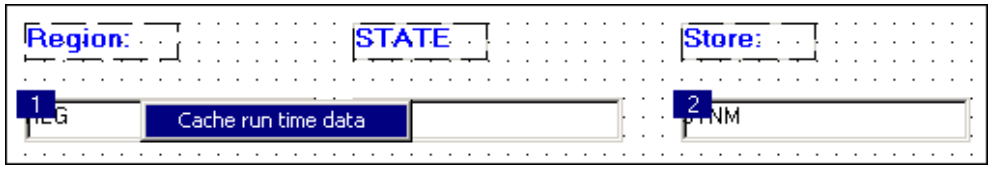
Procedure: How to Remove a Parameter From the Chain

1. Click the parameter you want to remove from the chain.
2. Click the button in the layout toolbar.
3. Press the shift key and click the remaining chained parameters.
4. Click the *Chain order* button in the layout toolbar. You no longer see a numeric label on the parameter that you removed.

Procedure: How to Cache Parameter Data

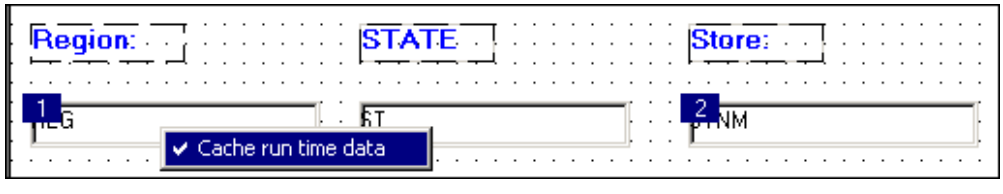
When creating a chained set of parameters you also have the option to "cache" the parameter data. Running a cached version of the parameter data makes the selection of parameters much quicker because the browser retrieves all the data initially from the data source and caches it on the client. By default, chained parameter data is non-cached, meaning every selection causes the parameter data to be reloaded from the server. However, once you make a selection, the values are cached. The next time you make that selection, the cached data is used.

1. Click a chained parameter in the layout.
2. Right-click the parameter and click the *Cache run time data* option.



You may select other chained parameters and enable the *Cache run time data* option.

Tip: If you decide to deactivate the option and revert back to using non-cached data, right-click the chained parameter and click the check mark next to the *Cache run time data* option.



For more information about parameters, see *Using Forms to Supply Parameter Values* in the *Designing a User Interface for a Web Application With the Resource Layout Painter* chapter in the *Developing Reporting Applications* manual.

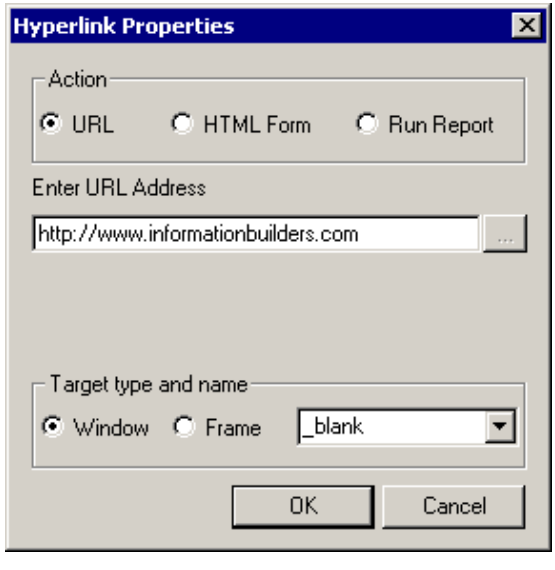
Adding an Image With a Hyperlink to the Layout

In Release 5.2.3, you can insert an image into your report layout and add a hyperlink to it. After you run your report and click the image, you can launch a URL or run a report the same way you can by clicking a hyperlink or push button.

Procedure: How to Add an Image With a Hyperlink

1. Open the Resource Layout Painter.
2. Do one of the following:
   - Click the *Insert image* button.
   - From the Insert menu, select *Image*.
   - Right-click in the layout, select *Insert* from the pop-up menu, then select *Image*.

   The Get source file dialog box opens.
3. Click the browse button (...) to the right of the blank field and select the image you want to add to the layout.
4. Click *OK*.
5. Right-click the image and select *Image Properties* and then *Create Hyperlink*. The Hyperlink Properties dialog box opens.
6. Ensure that the URL radio button is selected in the Action section of the Hyperlink Properties dialog box and type a name in the Enter URL Address field.



7. Click *OK*. Execute the request and click the image to launch the Web page of the URL address you typed in the Enter URL Address field.

Displaying Radio Buttons and Check Box Lists in Multiple Columns

Reference:

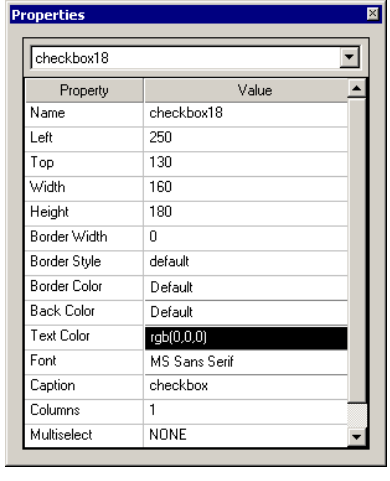
Check Box Properties Window

Radio Button Properties Window

In Release 5.2.3, you can display radio buttons and check box lists within a report layout in multiple columns. The Properties dialog boxes for radio buttons and check boxes has a Columns option in which you can change the number of columns used to display the controls.

You can also enter -1 (which represents an infinite number) if you want to display the controls horizontally with an unlimited number of columns. Use the Check Box Properties Window or the Radio Button Properties Window to use the Columns option.

Reference: Check Box Properties Window



The Check Box Properties Window contains the Columns option.

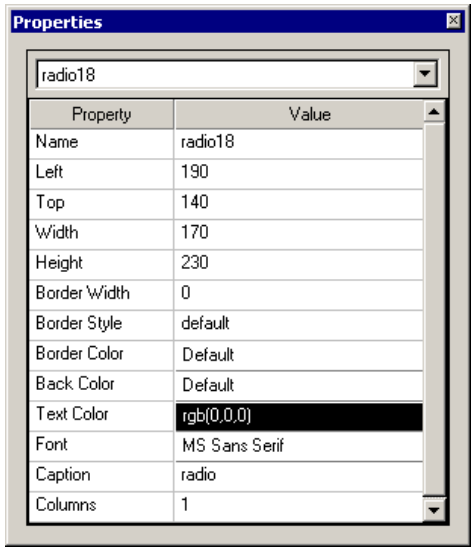
Columns

Determines the number of columns in which check boxes are displayed. 1 is the default value. Enter 2 or a higher number to display check boxes in multiple columns.

You can also enter -1 (which represents an infinite number) if you want to display the check boxes horizontally with an unlimited number of columns.

All options are fully documented in *Check Box Properties Window* in the *Designing a User Interface for a Web Application With the Resource Layout Painter* chapter in the *Developing Reporting Applications* manual.

Reference: Radio Button Properties Window



The Radio Button Properties Window contains the Columns option.

Columns

Determines the number of columns in which the radio buttons list is displayed. 1 is the default value. Enter 2 or a higher number to display the radio buttons list in multiple columns.

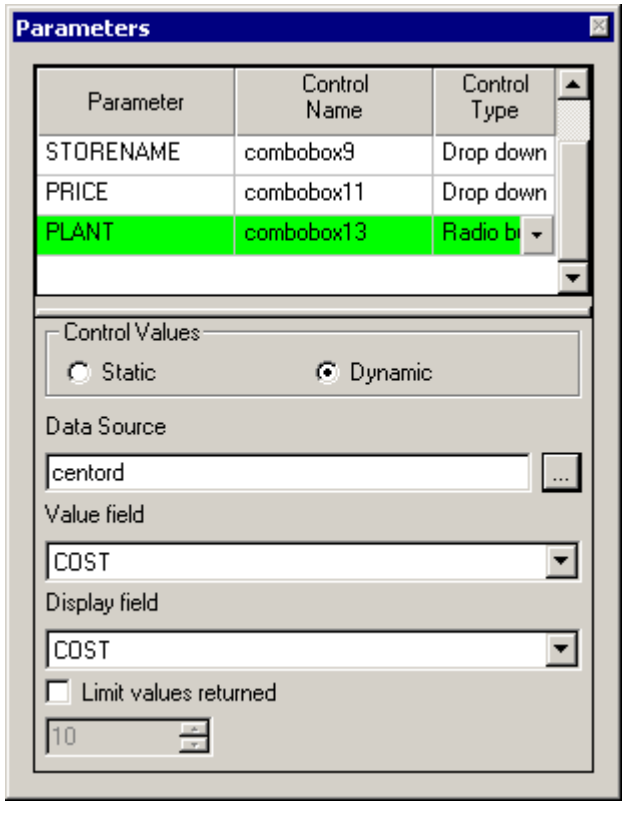
You can also enter -1 (which represents an infinite number) if you want to display the radio buttons horizontally with an unlimited number of columns.

All options are fully documented in *Radio Buttons Properties Window* in the *Designing a User Interface for a Web Application With the Resource Layout Painter* chapter in the *Developing Reporting Applications* manual.

Specifying Different Names for Dynamic Parameter Values

In Release 5.2.3, you can specify different names for a dynamic parameter's value in the Display and Value fields on the Parameters Window in the Resource Layout Painter. The parameter's value displays by default unless you specify a different name in the Display field. Use the Parameters Window (Dynamic list).

Reference: Parameters Window (Dynamic list)



The Parameters Window contains the Value field and Display field.

Value field

Is the data source field from which the values are retrieved.

Display field

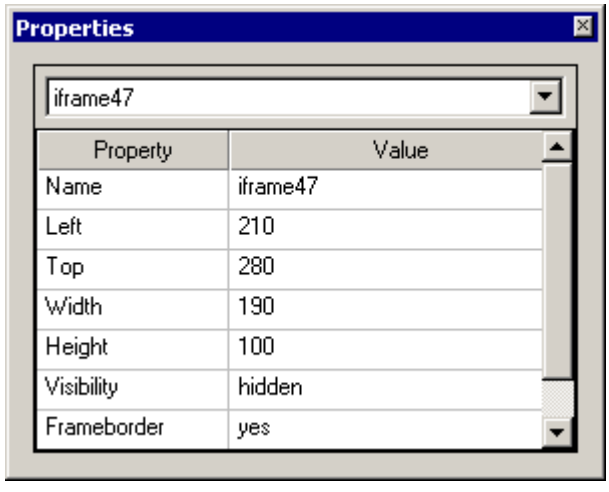
Is the text that represents the parameter value in the form the user views.

All options are fully documented in *Parameters Window (Dynamic list)* in the *Designing a User Interface for a Web Application With the Resource Layout Painter* chapter in the *Developing Reporting Applications* manual.

Hiding Borders Around Frames

In Release 5.2.3, you can hide borders around frames in the report layout using the Frame Properties Window.

Reference: Frame Properties Window



The Frame Properties Window contains the Frameborder option.

Frameborder

Determines whether or not a border appears around the frame when the request is executed. Yes is the default value. To hide borders around a frame, choose *no* from the drop-down list.

All options are fully documented in *Frame Properties Window* in the *Designing a User Interface for a Web Application With the Resource Layout Painter* chapter in the *Developing Reporting Applications* manual.

Applying Parameter Values to External Reports

In Release 5.2.3, you can apply parameters to external reports in a report layout the same way that you can apply them to reports you create with the Report Painter. You can also update and target parameters for an external report to a frame or window directly within your report layout. See your Information Builders representative for details.

Source File Management

Developer Studio supports third-party source file management products such as Microsoft Visual SourceSafe®, PVCS® Version Manager™ from MERANT™, and others that use an industry standard API. Source file management provides version control for individual projects and team projects, enabling multiple users of Developer Studio to work on the same files without overwriting each other's modifications. A source file management product typically controls access to source code, and documents all code changes.

From the Explorer, a developer can track the code for a project—for example, the procedures in a project—through an audit trail of changes.

This feature is available from the Projects on localhost area in Developer Studio and is not applicable to files in the WebFOCUS Environments area.

Procedures and guidelines for source file management are fully documented in *Managing a Project With Source Control* in the *Creating a Reporting Application* chapter in the *Developing Reporting Applications* manual.

Update Assist

In this section:

Dynamic List Generation
Results Page
Enhanced DBA Prompting
Enhanced Data Retrieval Error Messages
Tree Navigation Enhancements
Enhanced Date Field Validation
Validation of User-Supplied Data in Update Assist Applications
Choosing a Color Scheme for an Update Assist Application
Customizing Update Assist Designs
Calendar Control for Date-Formatted Fields in Update Assist
Right-Click Menu Options on Trees
Sorting Tables

The new Update Assist tool enables you to create fully functional applications by answering a few simple questions. The applications use Maintain and can perform a combination of add, update, delete, and search functions.

The Update Assist tool is ideal for generating rapid prototypes for any Maintain-based application. The tool uses standard templates. Applications generated with Update Assist can also be used as quick file browsers, and can generate single-screen updates that seamlessly work with WebFOCUS report drill downs. Update Assist-generated application screens can be easily inserted into WebFOCUS reports and layouts.

The Update Assist tool is fully documented in the *Creating an Update Application With Update Assist* chapter in the *Developing Reporting Applications* manual.

Dynamic List Generation

Update Assist provides you with the ability to generate dynamic lists in your applications. This means that you can add pick lists to your application which get their data from a database column or a flat file, without having to write any supporting code. With Dynamic Lists, any database column can be used as the source of pick lists on your forms.

To activate the feature for any field in your application, select the Dynamic List option from the Validation drop-down list in Step 2 of the Update Assist Wizard.

Results Page

Update Assist lets you specify if you want an alternative Results page to display once a user commits a Save or Delete action. This gives you the functionality to embed an Update Assist application in an existing Web application. When users complete a Save action, instead of seeing a message displayed in the Message Area near the top of the form, they are sent to the Results page, where they see a message ("Your changes have been saved.")

If you want your application to use a Results page, check the *Generate Results Page* box in the Update Assist's Navigation Options - Step 3 of 6 dialog box. When you have completed all steps in the wizard, four (instead of three) Maintain forms are added to your project. The Results page is created and sourced as a standard Maintain form, and can be customized and edited in the Maintain Development Environment.

You can customize the messages displayed in this form by editing the Maintain procedure. You can, for example, place a button or link on the Results page that takes a user to another location, or if the Maintain procedure was called from a report, navigates the user back to a rerun version of the report with the new data showing.

Tip: While you are able to generate the Results page regardless of which navigation option you choose, this option is most appropriate for the "no navigation" scenario. It is best to use the Results page with an application that calls the Maintain procedure directly with a URL and displays the Maintain form in a frame or iFrame of the application's workspace. When using the Tree, Combo, or Edit box navigation, users must renavigate to their information or click the browser's Back button to get back to a form with controls.

Enhanced DBA Prompting

You can mask passwords during DBA prompting, and you are reprompted if you enter an incorrect DBA password.

Enhanced Data Retrieval Error Messages

Any errors that occur during data retrieval are specifically listed inside the navigation pane. Such errors can occur if you do not place a Master File in an application path, if a data file is missing from an application, or if the WebFOCUS Reporting Server is not running.

Tree Navigation Enhancements

The Tree control used in Tree navigation displays in just a few seconds, even with database selections of tens of thousands of rows.

You can use two JavaScript variables that control the sub-foldering of nodes to customize the Tree control. Modify these variables by editing the *segname*_navbar.htm file (the navigation bar HTML file that is generated by Update Assist). It is in the HTML files directory for your project after you run Update Assist:

- **maxNodesPerFolder** is an integer value that controls the maximum number of nodes that can be displayed in each folder. If the number of nodes exceeds this number, the Tree control uses sub-folders labeled with a range of the node values contained within to automatically split up the nodes.
- **maxFolders** is an integer value that determines the maximum number of folders that can be rendered. The total number of nodes that are rendered are divided among these folders.

Note: If the maxNodesPerFolder and maxFolders values are set in a way that would cause a conflict, the setting for maxFolders takes precedence over maxNodesPerFolder.

Enhanced Date Field Validation

If your source database has date/time fields, Update Assist's automatic validation correctly handles validating user input for these fields. Also, partial date fields (for example, Month/Year, Month only, Year only) validate correctly.

Validation of User-Supplied Data in Update Assist Applications

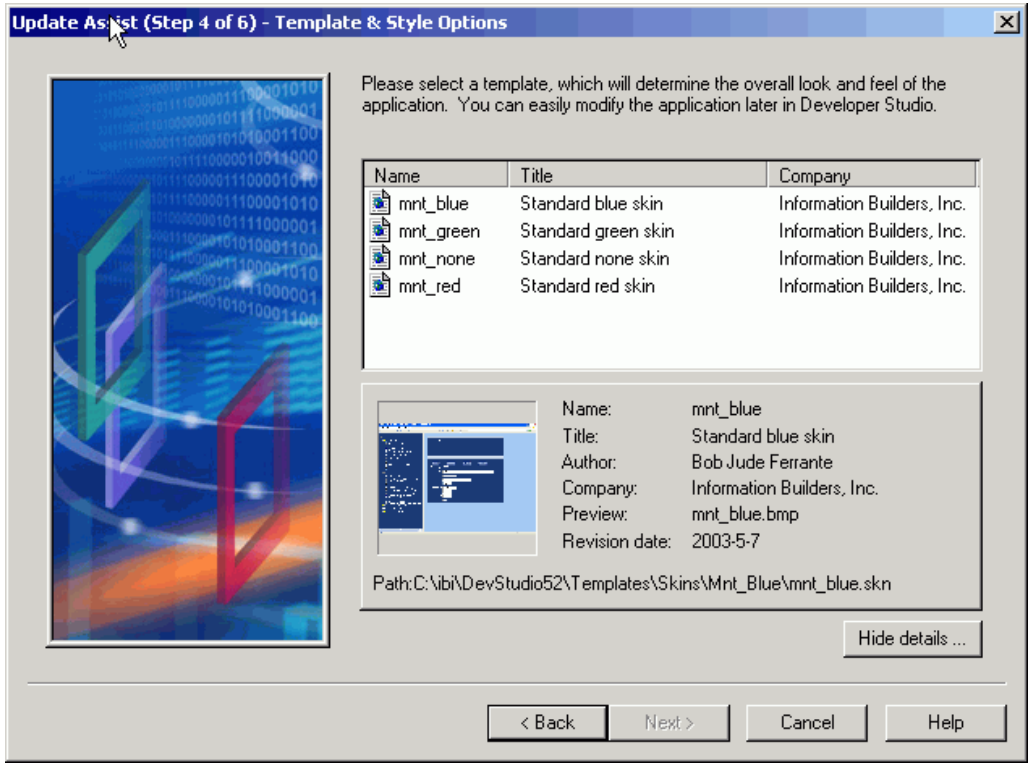
In Release 5.2.3, rather than performing validation as users enter data into each field, applications generated with Update Assist validate data only when the user submits the information, for example, by clicking *Save* if the form contains a Save button. If the form does not contain a Save button, data is submitted when the user clicks *Next* to go to the next step.

For procedures and examples, see *Update Assist (Step 2 of 6): Selecting Fields to Update* in the *Creating an Update Application With Update Assist* chapter in the *Developing Reporting Applications* manual.

Choosing a Color Scheme for an Update Assist Application

In Release 5.2.3, Update Assist has six windows in which to create applications. The fourth window of Update Assist is where you select the color scheme for your Update Assist application.

In the fourth window, select a color scheme and click *Next*. To see a preview of the color scheme and more information about it, select the particular color scheme and click *Show details*.



For complete Update Assist steps, see the *Creating an Update Application With Update Assist* chapter in the *Developing Reporting Applications* manual.

Customizing Update Assist Designs

In Release 5.2.3, by using a new XML-based definition language, developers can create custom looks for applications generated with Update Assist. They can add corporate logos and Cascading Style Sheets (CSS) to create WebFOCUS Maintain applications that all share a common corporate look and feel. For technical documentation on how to use this feature, contact your Information Builders representative.

Calendar Control for Date-Formatted Fields in Update Assist

In Release 5.2.3, a calendar icon appears next to changeable date-formatted fields. When a user clicks the calendar icon, a calendar appears; any date selected on this calendar is entered into the date field. Users can also enter dates into the date field manually.

Right-Click Menu Options on Trees

If you have selected *Key values selected via tree* in Step 3 of the Update Assist Wizard, your update application generates a form in which the user can select records using a hierarchical tree control.

In Release 5.2.3, users can perform update functions directly from the tree by right-clicking the tree item and selecting one of the options:

- *Select* opens the record in the form on the right side of the application.
- *Delete* removes the record from the data source.
- *New* enables the user to add a new record to the data source.
- *Search* enables a user to search for records.

Note: Right-click menu options depend on the properties of your Update application. For example, if you did not select Add in Step 2 of the Update Assist wizard, the New option would not appear on the right-click menu.

Sorting Tables

In Release 5.2.3, when you perform a search in your Update application, you can sort the results in ascending or descending order by clicking a column heading.

These sorting options are available for all of your application tables. For more information, contact your Information Builders representative.

Financial Report Painter Enhancements

In this section:

Reporting Dynamically From a Hierarchy

Indenting Row Titles

Displaying an FML Hierarchy With the Financial Report Painter

Consolidating Data in a Hierarchy in Developer Studio

Formatting Enhancements

How to:

Adjust Indents for Captions in a Hierarchy

Format a Column in the Design Matrix

Format a Row in the Design Grid

Format a Cell in the Design Matrix

Example:

Displaying an FML Hierarchy With Captions

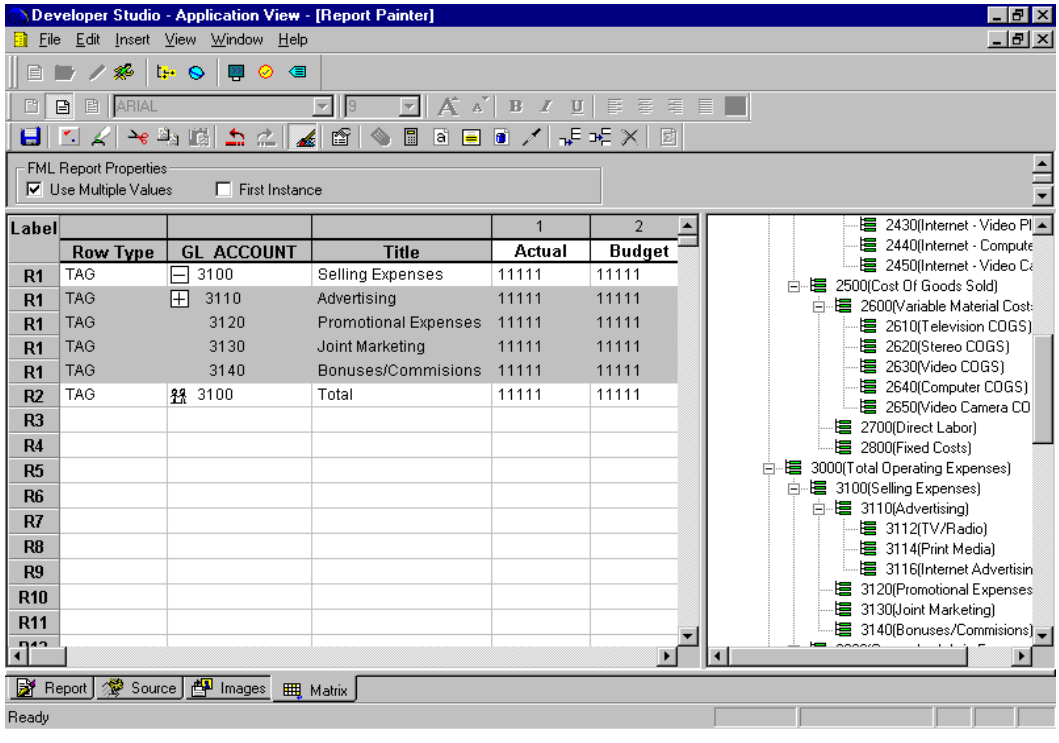
Consolidating Data in a Hierarchy Into One Line of Report Output

The Financial Report Painter and the underlying Financial Modeling Language (FML) have been enhanced to support reporting against a hierarchical data structure, more flexible use of fields in a request, and rows and cell styling from the Financial Report Painter. You can access the Financial Report Painter through the Report Painter.

This tool is documented in the *Creating Reports With Financial Report Painter* chapter in the *Creating Reports With Graphical Tools* manual, and in the *Creating Financial Reports* manual, which includes tutorials. The FML language is fully documented in the *Creating Financial Reports* chapter in the *Creating Reports With WebFOCUS Language* manual.

Reporting Dynamically From a Hierarchy

In the Financial Report Painter, the hierarchy defined in the Master File is reflected in the FOR field values panel to the right of the Design matrix, as shown in the following illustration. If the hierarchy fields are defined with captions in the Master File, the resulting report indents the captions proportional to their levels in the hierarchy.



The Financial Report Painter supports most of the row styling capabilities that are available in the WebFOCUS language. In addition, you can format an individual cell. Drill down functionality is also supported at the cell level.

Formatting of columns, rows, and cells is managed through the familiar Report Painter Field Properties and Report Options Style tabs, which can be accessed directly from the Financial Report Painter.

Indenting Row Titles

To clarify relationships within an FML hierarchy, the captions (titles) of values are indented at each level.

For language implementation of this feature, see *Indenting Row Titles in an FML Hierarchy* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.

Procedure: How to Adjust Indents for Captions in a Hierarchy

If you wish to adjust the indents:

1. Choose *Indent* from the File menu. The Indent dialog box opens.

   If a hierarchy has not been defined in the Master File for the data you are reporting against, the Indent option does not appear.

2. Choose one of the following radio buttons:

   - *Turn indent off*. This option left-justifies titles for values at all levels of the hierarchy.
   - *Select a value of indent*, then enter an explicit measurement to represent the number of spaces to be indented based on the number of blank spaces preceding the caption text in the Master File and the unit of measurement defined in the Page Setup dialog box (inches, centimeters, or points). For example, suppose that the caption text is preceded by two blanks and the base measurement is inches. If you enter .2, each level of values in the hierarchy is indented .2 inches for each leading space (.2" x 2) from the previous level.

Displaying an FML Hierarchy With the Financial Report Painter

When reporting from an FML hierarchy, you can dynamically retrieve and display hierarchical data using two variations:

- *Show only children* displays only the children, not the parent value for those children.
- *Show with children* displays the parent and then the children.

In either case, you can show all children for the specified parent, or children to a specified level in the hierarchy, up to 99 levels deep. The default hierarchy depth is one level. To highlight these relationships visually, successive levels of the hierarchy field are indented two spaces from the previous level.

The hierarchy is displayed sorted by the parent field and, within the parent, sorted by the hierarchy field.

When displaying a hierarchy, you can show either detail or summary data for fields other than the For field in the request. You can also replace the For field value with a descriptive caption.

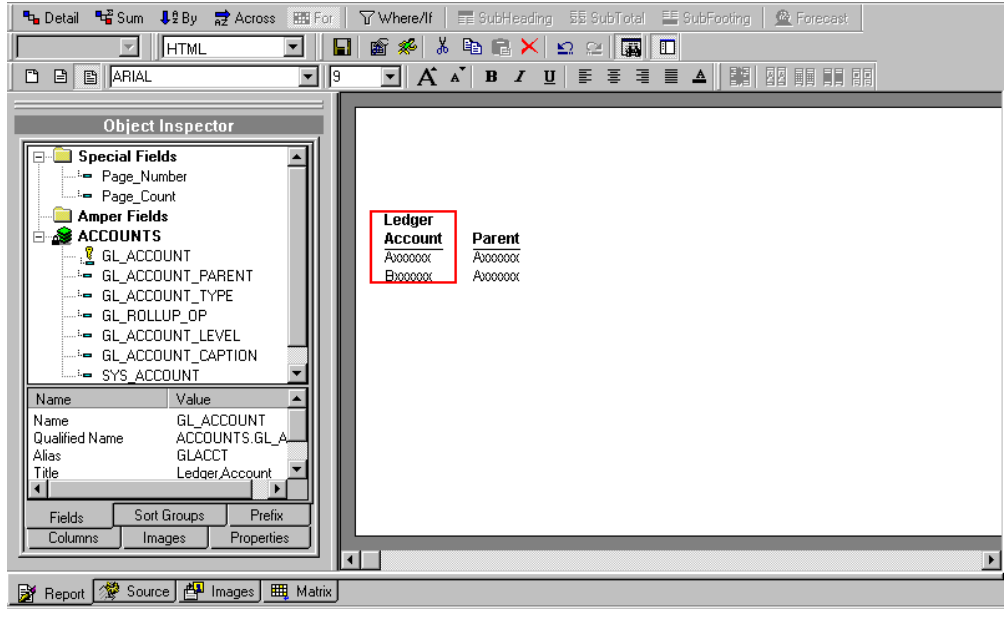
For language implementation of this feature, see *Displaying a Dynamic FML Hierarchy* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.

Example: Displaying an FML Hierarchy With Captions

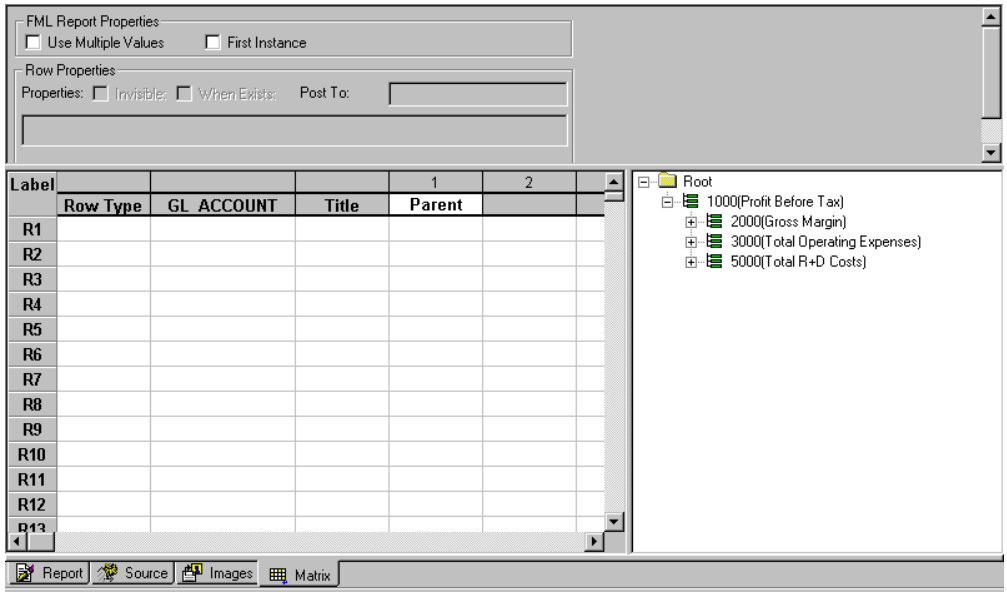
The following example displays two levels of a charts of accounts hierarchy and shows descriptive captions defined in the Master File, instead of the account numbers, in the report.

1. Create a report procedure in Developer Studio and name the file *fmlhiercaps*.
2. When prompted, choose Report Painter as your reporting tool (it is the default) and choose CENTGL as your data source.

   The Report Painter opens with CENTGL fields listed in the Object Inspector panel.
3. Drag the field GL_ACCOUNT_PARENT into the Report Painter window; the column title associated with this field is Parent. Select *Parent* and click the *Detail* button.
4. Now drag GL_ACCOUNT into the Report Painter window; the column title associated with GL_ACCOUNT is Ledger Account. Select *Ledger Account* and click the *For* button. (This becomes the controlling field in your financial report.) The Report Painter looks like this:



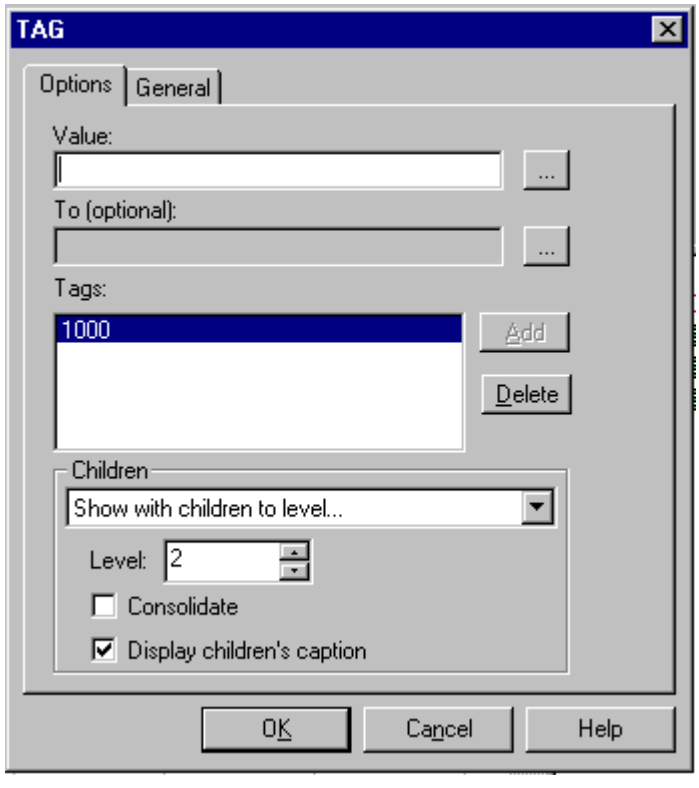
5. Click the Matrix tab below the Report Painter to open the Design matrix. Note that GL_ACCOUNT (the For field) is the title of the second column and its values appear in the For field values panel at the right of the matrix. You will be populating the matrix with these values.



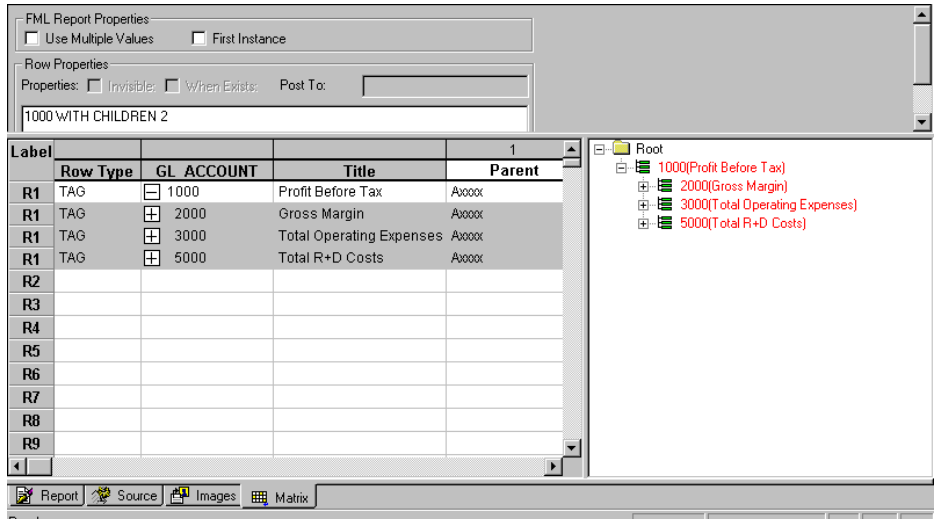
6. Begin by dragging the tag 1000 from the FOR field values panel into row R1 of the matrix. The 1000 account tag appears in the GL_ACCOUNT column.

7. In row R1, right-click 1000 and select Row Properties from the menu. The TAG dialog box opens, with 1000 listed in the Tags box.

8. In the Children box, choose *Show with children to level...* from the drop-down list, then select *2* in the Level box to display two levels of the hierarchy, with account 1000 as the starting point (or parent level). The *Display children's captions* check box is selected by default. This shows the descriptive titles for the children, rather than their tag values (1000, 2000, and so on), on the report. The dialog box looks like this:



9. Click the plus (+) sign next to 1000 to expand the hierarchy one level. The matrix now looks like this:



If you wish, repeat the process to expand the hierarchy another level.

10. To add some quick styling, click the *Report Options* icon on the toolbar above the matrix. On the Style tab, click the *Style File Selection* button (bottom right of window). The Style File Selection dialog box opens. Select the style sheet *DefaultGrid (DEFFLT1.STY)* under Style Sheet Source. Click *OK* to confirm your choice and *OK* again to return to the matrix.

11. Click the *Run* icon on the toolbar to see the report, which lists the account numbers associated with the levels of the parent/child hierarchy. The indents for the hierarchy levels are set by default.

	Parent
Profit Before Tax	
Gross Margin	1000
Sales Revenue	2000
Cost Of Goods Sold	2000
Total Operating Expenses	1000
Selling Expenses	3000
General + Admin Expenses	3000
Total R+D Costs	1000
Salaries	5000
Misc. Equipment	5000

Tip: If you wanted to see the children in the hierarchy without the parent, you could choose *Show only children to level 2* in step 8, rather than *Show with children to level 2*. Without the parent line, the report would look like this:

	Parent
Gross Margin	1000
Sales Revenue	2000
Cost Of Goods Sold	2000
Total Operating Expenses	1000
Selling Expenses	3000
General + Admin Expenses	3000
Total R+D Costs	1000
Salaries	5000
Misc. Equipment	5000

Consolidating Data in a Hierarchy in Developer Studio

The Consolidate option consolidates multiple levels of the hierarchy on one line of the report output. Consolidate can be used alone or in conjunction with the *Show only children* or *Show with children* options. Consolidation is designed to work with requests that use the Sum option for fields other than the For field. It is also designed to be used with detail-level financial data, not data that is already consolidated.

- When used *alone*, the Consolidate option aggregates the parent and children on one line of the report output, summing the numeric data values included on the line.
- When used in conjunction with *Show only children*, the Consolidate option displays one line for each child of the specified parent value. Each line is a summation of that child and all of its children. You can specify the number of levels of children to display (which determines the number of lines generated on the report output) and the depth of summation under each child. By default, only direct children have a line in the report output and the summary for each child includes all of its children.
- When used in conjunction with *Show with children*, the Consolidate option first displays a line in the report output that consists of the summation of the parent value and all of its children. Then it displays additional lines identical to those displayed by *Show only children* plus *Consolidate*.

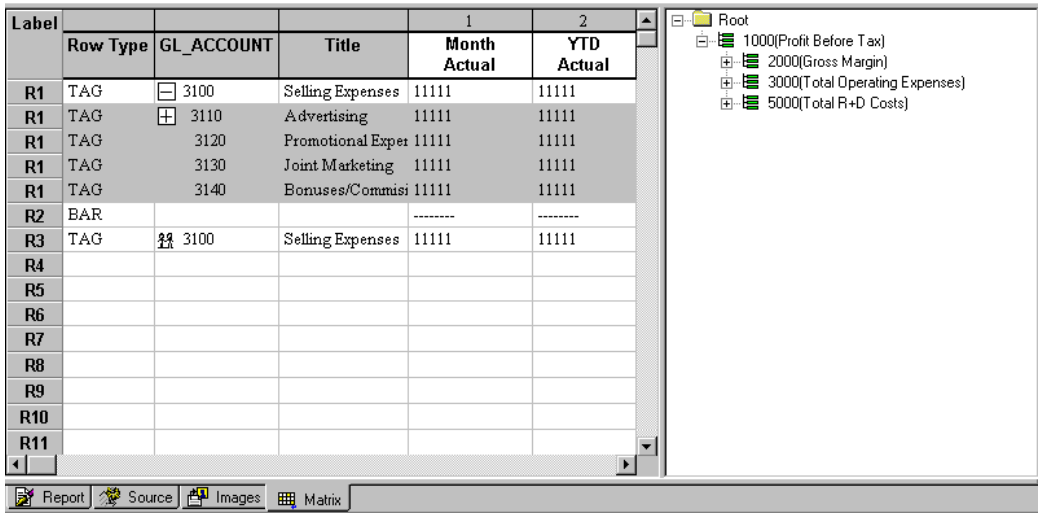
In order to use a data record in more than one line of a financial report (for example, to display both detail and summary lines or to consolidate detail data at multiple levels), select the option *Use Multiple Values* before you begin to populate the matrix.

When using WebFOCUS instead of Developer Studio, use the SET FORMULTIPLE command to use a data record in more than one line of an FML report. Using the SET FORMULTIPLE command, you can include the same value of a FOR field— whether as a single tag value, as part of a range, or in a calculation— in multiple rows of the FML matrix. This feature is available in the Financial Report Painter simply by clicking the *Use Multiple Value* check box above the Design matrix.

For language implementation of this feature, see *Consolidating Data in an FML Hierarchy* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.

Example: Consolidating Data in a Hierarchy Into One Line of Report Output

The following matrix displays detail data for Selling Expenses, the parent level in a hierarchy, and its children, followed by a summary line for all Selling Expenses.



Label				1	2
	Row Type	GL_ACCOUNT	Title	Month Actual	YTD Actual
R1	TAG	3100	Selling Expenses	11111	11111
R1	TAG	3110	Advertising	11111	11111
R1	TAG	3120	Promotional Expes	11111	11111
R1	TAG	3130	Joint Marketing	11111	11111
R1	TAG	3140	Bonuses/Commisi	11111	11111
R2	BAR			--------	--------
R3	TAG	3100	Selling Expenses	11111	11111
R4					
R5					
R6					
R7					
R8					
R9					
R10					
R11					

It produces the following output:

	Month Actual	YTD Actual
Selling Expenses	.	.
Advertising	.	.
TV/Radio	1,049,146.	2,954,342.
Print Media	244,589.	721,448.
Internet Advertising	9,542.	29,578.
Promotional Expenses	53,719.	151,732.
Joint Marketing	97,135.	289,799.
Bonuses/Commisions	100,188.	304,199.
Selling Expenses	1,554,319.	4,451,098.

What makes this reporting dynamic?

Suppose that you run this report weekly. If you ran the report today, and tomorrow another account were added as a child of 3100, next week's report would automatically reflect the change in the hierarchy, as well as changes to the data, based on the versatile parent/child declaration in the Master File. No adjustment would be needed to keep either the Master File or the report request up to date.

Formatting Enhancements

You can apply a wide range of formatting options to individual columns, rows, and cells in a financial report using options on the Field Properties Style tab. You can further refine formatting:

- For columns, by identifying the column title and column data as separate objects for styling.
- For rows, by identifying the row and title as separate objects for styling.

If you style a row and then a cell within that row, the cell styling takes precedence for the cell.

If you style a column and then a cell with that column, the cell styling takes precedence for the cell.

If you style a cell, then style a row or column that the cell is in, the cell style remains the same.

Row and cell styling options are specific to financial reports created in the Financial Report Painter. Column styling options are identical whether applied from the Financial Report Painter or the Report Painter.

For language implementation of this feature, see *Formatting an FML Report* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.

Procedure: How to Format a Column in the Design Matrix

1. Right-click a column and choose *Options* from the shortcut menu. The Field Properties dialog box opens at the Style tab.
2. Select *Column Title and Data* as the active object, or select *Column Data* or *Column Title* to style them separately.

   If you have already styled either the title or data for a column and wish to quickly apply the same formatting to the other element, click the appropriate button: *Copy Title Style to Data* or *Copy Data Style to Title*.
3. Under Column Layout, you can change column width and justification.

   Width options are:

   Maximum

   Sets the width according to the length defined in the field format.

   Minimum

   Sets the width of the column according to the widest value or heading in the field. Minimum is the default selection.

   Truncate

   Enables you to specify where to truncate the column width based on the specified units (inches or centimeters). To indicate that a field value has been truncated in the browser, WebFOCUS places an exclamation point (!) after every alphanumeric and text field value and an asterisk (*) after every numeric field value.

   Wrap

   Enables you to specify where to wrap data based on the specified units (inches or centimeters).

   Justification options are: left, right, center, or return to the default positioning.

4. Under Graphical options, select the font characteristics, border or grid characteristics, and/or background colors that you wish to apply to the column:

   a. **For borders,** click the *Select Borders* button. The Borders dialog box opens. Select width, style, and/or color options from the drop-down lists. Click *OK*.

      You can apply the same specifications to all border lines, or vary specifications for top, bottom, right, and/or left borders.

      Note: To set borders in an HTML report, you must turn Cascading Style Sheets On. Click the *Report Options* icon on the toolbar and select the Output tab. Under *Display Options*, select HTML and verify that On is selected from the Cascading Style Sheets drop-down menu.

      When Borders is selected, Grids is disabled.

   b. **For grids,** click the *Select Grids* button. Select a line style and indicate whether to display horizontal lines, vertical lines, or both. Click *OK*. This option applies to columns in PDF reports; it does not apply to columns in HTML reports.

      When Grids is selected, Borders is disabled.

   c. **For fonts,** click the *Select Fonts* button. The Fonts dialog box opens. Select font name, font style, font size, and/or color. Click *OK*.

   d. **For background color,** select the *Single Color* radio button under Background Coloring, and choose a color from the palette. Click *OK*.

      If you identify the active object as column data, the *Alternating Background Colors* button is activated. You can use this feature to assign colors to alternating rows in one or more columns.

5. Under *Applying to condition* in the Style tab, you can define or edit a condition that controls when specified formatting options are applied to one or more columns.

6. Click *OK* to return to the Design matrix where many styling changes are reflected.

7. Click the *Run* icon on the toolbar to see the column formatting options applied in the report output.

Tip: If you wish to affect other column features, click the:

- **Drill Down tab** to drill down to another procedure, a URL, or another supported option.
- **General tab** to change the column title or field format, to make the column visible or invisible in the output, or to activate other features.

For details about defining conditional report styles and other column formatting features, see the *Styling Reports With the Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.

Procedure: How to Format a Row in the Design Grid

1. Right-click a row label (either a default label like R1 or an explicit row label that you have assigned) and select *Options* from the shortcut menu. The Field Properties dialog box opens at the Style tab, with Row identified as the active object.
2. Under *Graphical*, at the right of the Style tab, select the font characteristics, border characteristics, and/or background colors that you wish to apply to the row.
   a. **For borders,** click the *Select Borders* button. The Borders dialog box opens. Select width, style, and/or color options from the drop-down menus.

      You can apply the same specifications to all border lines, or vary specifications for top, bottom, right, and/or left borders. Click *OK*.

      Note: To set borders in an HTML report, you must turn Cascading Style Sheets On. Click the Report Options icon on the toolbar and select the Output tab. Under Display Options, select HTML and verify that On is selected from the Cascading Style Sheets drop-down menu.

   b. **For fonts,** click the *Select Fonts* button. The Fonts dialog box opens. Select font name, font style, font size, and color. Click *OK*.

   c. **For background color,** select the *Single Color* radio button under Background Coloring and choose a color from the palette. Click *OK*.
3. Click *OK* to return to the Design matrix, where many styling changes are reflected.
4. Click the *Run* icon on the toolbar to see the row formatting options applied to the report output.

Procedure: How to Format a Cell in the Design Matrix

1. Right-click a cell and choose *Options* from the shortcut menu. The Field Properties dialog box opens at the Style tab, with *Cell* identified as the active object.
2. Under *Graphical*, select the font characteristics, border characteristics, and/or background colors that you wish to apply to the cell:
   a. **For borders,** click the *Select Borders* button. The Borders dialog box opens. Select width, style, and/or color options from the drop-down menus. Click *OK*.

      You can apply the same specifications to all border lines or vary specifications for top, bottom, right, and/or left borders.

      Note: To set borders in an HTML report, you must turn Cascading Style Sheets On. Click the *Report Options* icon on the toolbar and select the Output tab. Under Display Options, select *HTML* and verify that *On* is selected from the Cascading Style Sheets drop-down menu.

   b. **For fonts,** click the *Select Fonts* button under Graphical options. The Fonts dialog box opens. Select font name, font style, font size, and color. Click *OK*.

   c. **For background color,** select the *Single Color* radio button under Background Coloring, and choose a color from the palette. Click *OK*.
3. Under *Applying to condition*, you can define or edit a condition that controls when specified formatting options are applied to a cell.
4. Click *OK* to return to the Design matrix, where most formatting changes are reflected.
5. Click the *Run* icon on the toolbar to see the cell formatting options applied in the report output.

Note: If you wish to drill down from the selected cell to another procedure or to a URL, click the Drill Down tab. For details, see *Drilling Down From a Column, Cell, or Row* in the *Creating Reports With Financial Report Painter* chapter in the *Creating Reports With Graphical Tools* manual.

Although certain options on the General tab are enabled when *cell* is the active object, these options are applied to the entire column, not only to the selected cell.

SET Parameters

You can issue a SET command by using the SET tool to generate the command in a stored procedure, or by coding it in a stored procedure. If you use the SET tool, a SET command is generated for the report you are developing. SET commands are fully documented in the *Customizing Your Environment* chapter the *Developing Reporting Applications* manual. The following SET commands are supported in Developer Studio.

- CDN. Enables you to choose how to punctuate numbers, using a combination of commas, decimals, spaces, and single quotation marks. For language implementation of this feature, see *Punctuating Numbers* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.
- CENT-ZERO. Displays a leading zero in decimal-only numbers. For language implementation of this feature, see *Displaying a Leading Zero in Decimal Numbers* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.
- DEFINEs. Compiles expressions into machine code for faster processing. For language implementation of this feature, see *Compiling Expressions Containing DEFINEs* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.
- GRAPHSERVURL. Creates a GIF file of the graph by sending an HTTP request to a servlet. For language implementation of this feature, see *Graph Enhancements* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.
- HDAY. Specifies the holiday file from which to retrieve dates that are considered holidays. For language implementation of this feature, see *Specifying Holidays* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.
- PCOMMA. Enables the reading of PC-type comma-delimited files with double quotation marks around alpha data and a carriage return line at the end of each record. For language implementation of this feature, see *Output Formats for Application Processing* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.

You can issue the following commands from the Financial Report Painter.

- BLANKINDENT. Clarifies relationships within an FML hierarchy by indenting the captions (titles) of values at each level. For language implementation of this feature, see *Indenting Row Titles in an FML Hierarchy* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.
- FORMULTIPLE. Enables you to include the same value of a FOR field in multiple rows of the FML matrix. For language implementation of this feature, see *Consolidating Data in an FML Hierarchy* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.

Functions for Displaying Fields

You can use three functions to adjust the display of fields in a report:

- SQUEEZ reduces multiple blank spaces to a single space.
- STRIP removes a character.
- TRIM removes leading or trailing occurrences of a pattern.

You can select these functions from the Functions tab in the Define tool.

For language implementation of this feature, see *Adjusting Field Displays* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.

CHAPTER 3

Managed Reporting Enhancements

This chapter describes all of the features for Managed Reporting in Version 5 Release 2 and has been updated to reflect Version 5 Release 2.3. Topics that contain Version 5 Release 2.3 features are marked with asterisks.

Topics:

User Administrator

* Developer User Type

* Managed Reporting Developer for Windows

* Change Management

* Security

Updated Presentation

Reporting Object Component Validation

New Create Synonym Behavior

Multiple Servers and Application Path

Custom Reports in Managed Reporting

* Execution of a Custom Report Using -INCLUDE

Deferred Receipt

Migration Utility

Repository Extract Utility

Dashboard Features

Creating Multiple Content Pages

Creating a Group View

Unique Customizations for Public Views

Removing the Banner

Customizing Content Block Layouts

Saving Most Recently Used Reports

Creating a My Favorite Reports List

Creating Custom Toolbars

Stand-Alone Domain/Role Trees

Optionally Displaying Frame Separators

Hiding the Display of Content Blocks

Default Public User

Domain Searches

User Options Settings

* Custom Reports in Dashboard

* Filters For Standard Reports in Dashboard

* Content Pages: ReportCaster and Report Library

* Dynamic Tracing

User Administrator

You can create a new user and assign the user's properties and password all in one step.

The User Administration User Properties window has been reorganized to be more intuitive and accommodate several additional properties.

The user privilege, User, represents the entry UAS User that can only run reports and not invoke Report or Graph Assistant, or save output from the OLAP slice-n-dice options. In addition, the Java User privilege has been renamed as Analytical User.

The public user and group required for Business Intelligence Dashboard are defined in the default Managed Reporting Repository.

Two capabilities have been created as follows:

- The ReportCaster Administrator is enabled when ReportCaster is installed and the user has Managed Reporting Administrator privileges.
- The Report Library option is available when ReportCaster and Report Library are installed.

The E-Mail Address field is an optional user property. The Push Notification through Email feature of the Report Library uses this property.

Developer User Type

In Release 5.2.3, the term "Domain Admin" has been replaced with "Developer." This more accurately reflects the role of this type of user, which is to develop content for Managed Reporting and not to administer users or groups. The internal flag for Developer remains *domadmin* for compatibility with previous releases.

Managed Reporting Developer for Windows

In Release 5.2.3, WebFOCUS Managed Reporting Developer for Windows enables Developers to create and manage domain content. Using this product, Developers can:

- Create Standard Reports and Reporting Objects with Windows-based tools such as the Report Painter.
- Analyze data with OLAP.
- Create and manage WebFOCUS Reporting Server resources such as metadata, procedures, and other files.

By design, all Managed Reporting administrative features are disabled. That means that Developers cannot create, delete, or rename Domains using the Managed Reporting Developer. There is also no access to User Administrator, Dashboard View Builder, or the ReportCaster Console. The Web Applications and User Management nodes have been removed in Explorer since they do not apply to this product.

Important: Managed Reporting Administrators can use this product, but their behavior is the same as that of Developers who have the Data Server privilege. Likewise, Developers still have their administrative capabilities when they use the full Developer Studio. Administrative functions are fully described in the *WebFOCUS Managed Reporting Administrator's Manual*.

Managed Reporting Developer for Windows can be used to create self-service applications. You can create Server procedures and metadata using the Data Server node (as long as you are either an Administrator or a Developer with the Data Server privilege). You cannot access the Web Applications folder from Explorer (which you might want to do to create launch pages) and you cannot work in the Project development paradigm which includes support for source control and deployment scenarios. In these cases, you must use the full Developer Studio.

The functionality of the Managed Reporting Developer for Windows product is described in the *WebFOCUS Managed Reporting Developer's Manual*.

Change Management

Developer Studio supports copy and paste of Managed Reporting resources between Managed Reporting repositories in order to help developers better manage their development, test, and production environments.

Developers with proper permission can multi-select reports, reporting objects, other files, folders, and sub-folders, or they can select a single domain. The internal name of each resource is checked, and a replace/add/merge dialog box is presented, as appropriate.

Many organizations do not grant Developers write access to the user acceptance test and production environments. Access to these environments is controlled and granted only to Administrators and/or automated change management processes. Yet only Developers know which changes are ready to be moved into test. In Release 5.2.3, the Change Management Extract Interface presents Developers with a graphical view of the Managed Reporting domains they manage and allows them to build a change package. This package is then loaded into another environment by the Change Management Load Program.

Change Management is fully documented in the *Change Management* chapters in the *WebFOCUS Managed Reporting Administrator's Manual* and the *WebFOCUS Managed Reporting Developer's Manual*.

Security

Managed Reporting security features include:

- Managed Reporting Servlet Exit
- Integrating Managed Reporting Security With Basic Web Server Authentication
- LDAP Authentication

Servlet Exit

The Managed Reporting Servlet Exit may be used for:

- **Authentication** - Authenticate Managed Reporting credentials outside of Managed Reporting/Dashboard. User IDs and privileges are still managed using the Managed Reporting User Administration tool. This is often referred to as the MRCEXT4 exit.

or

- **Authentication and Authorization** - Authenticate Managed Reporting credentials outside of Managed Reporting/Dashboard. Assign the following authorization:
  - User type (for example, Managed Reporting Administrator or Developer).
  - Privileges (for example, Shared and Schedule).
  - Domains (specify the domains to which the user has access).

  This is often referred to as the MRCEXT5 exit.

In Release 5.2.3, the Managed Reporting Servlet Exit may be used with ReportCaster. It uses a trusted connection between Managed Reporting and ReportCaster.

For detailed information about how to use the Managed Reporting Servlet Exit, see *Managed Reporting Servlet Exit* in the *WebFOCUS Custom Security Exits* chapter in the *WebFOCUS Security and Administration* manual.

Integrating Managed Reporting Security With Basic Web Server Authentication

To implement a single signon environment where the Web server identifies users, you can use Managed Reporting signon integration with Basic Web server authentication. In Release 5.2.3, Developer Studio, Dashboard, ReportCaster, and the HTML Report Assistant are fully supported when using this signon integration technique.

For more information about this technique, see the *Integrating WebFOCUS Security With Basic Web Server Authentication* chapter in the *WebFOCUS Security and Administration* manual.

LDAP Authentication

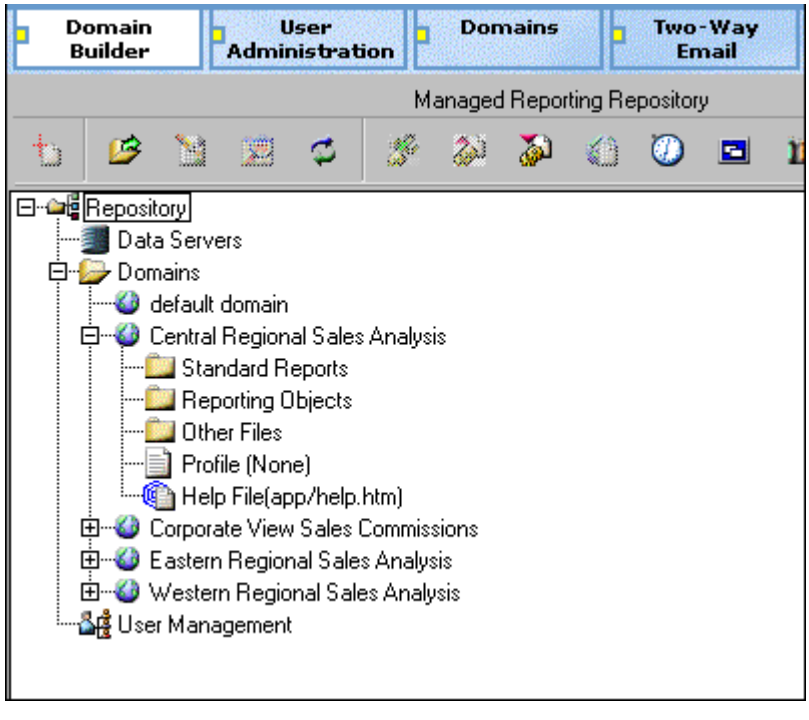
In Release 5.2.3 on Windows, Managed Reporting and ReportCaster users can use LDAP authentication. When using LDAP authentication, passwords are not defined, stored, and administered within Managed Reporting. Instead, Managed Reporting and ReportCaster are configured to authenticate users against passwords defined in an LDAP Directory.

You can configure LDAP Authentication during the WebFOCUS installation, as explained in the *Installing the WebFOCUS Client* chapter in the *WebFOCUS and ReportCaster Installation and Configuration for Windows* manual. After installation, a Managed Reporting Administrator adds users to Managed Reporting who are defined in an LDAP directory. Because Managed Reporting will authenticate the users against the LDAP Directory, the Administrator should not define passwords for the users in Managed Reporting.

Updated Presentation

A new look has been designed for WebFOCUS and Information Builders products that is consistent with the new ReportCaster and HTML Report Assistant. The windows are much less cluttered, and you can specify placement of the tabs across the top of the page or vertically on the left side.

The graphic below shows the Managed Reporting window with tabs placed horizontally.



Reporting Object Component Validation

Reporting Object components are validated prior to run time. Therefore, the process of finding and correcting errors is easier, since errors in individual components are detected prior to run time (whereas after run time, all components must be checked).

New Create Synonym Behavior

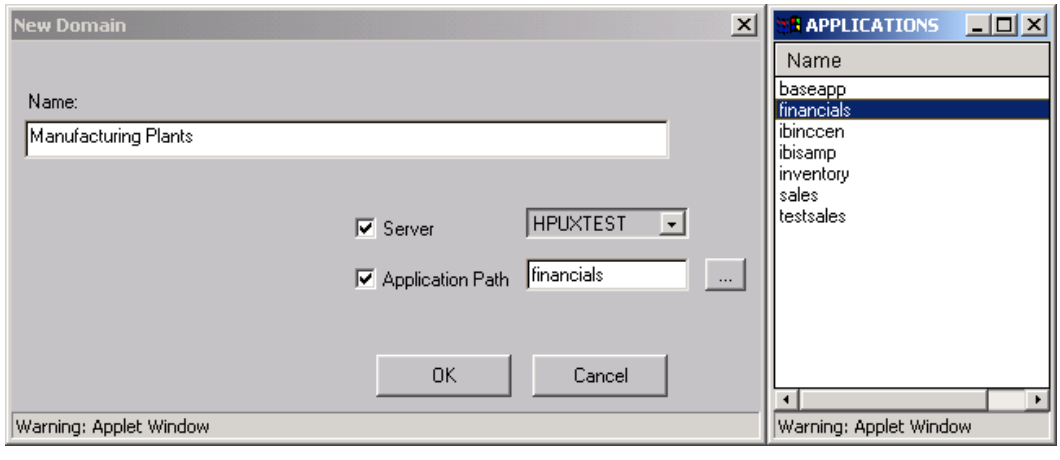
You create synonyms using the Server Console. The Server Console is accessed in the Data Server component when selecting Create Synonym. This functions for all supported platforms except for z/OS and OS/390 (MVS) (which still accesses the Synonym Wizard). Developers can view existing and new Master and Access Files and delete and refresh synonyms in the Server Console. Developers can also create synonyms for remote Reporting Servers and for relational and multidimensional data sources.

Multiple Servers and Application Path

You can choose the server on which you execute your reports. You accomplish this by checking the server property on a domain, report, or reporting object, and selecting it from a list.

You can also override the server's default search behavior by setting the Application Path property. This allows you to select or enter the application(s) that are searched by the server.

Note: You must have the Data Servers privilege in order to change the Server or Application Path.



The Data Servers feature is fully documented in the *Data Servers Feature* chapter in the *WebFOCUS Managed Reporting Developer's Manual*.

Custom Reports in Managed Reporting

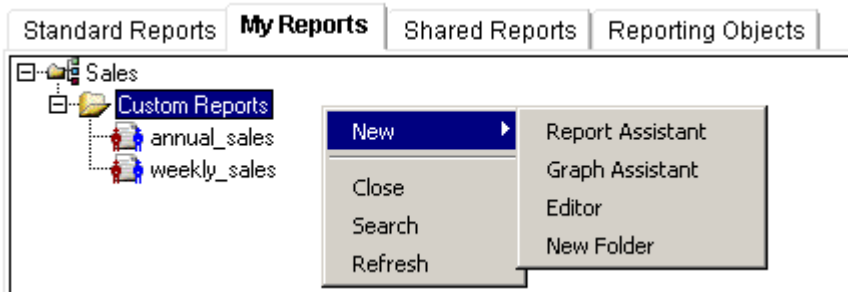
End users can edit existing reports, or create new reports called Custom Reports. Users can run, run deferred, share, and schedule Custom Reports, just like other reports.

Custom Reports are located in the My Reports tab in a folder called Custom Reports. This gives an Analytical user, Administrator, or Developer the capability of creating a new report with the assist tools or with an editor.

To gain this capability, the Administrator must grant the user the "Advanced" privilege. Administrators automatically have this new privilege and can assign it to Analytical users and Developers in the User Administration interface.

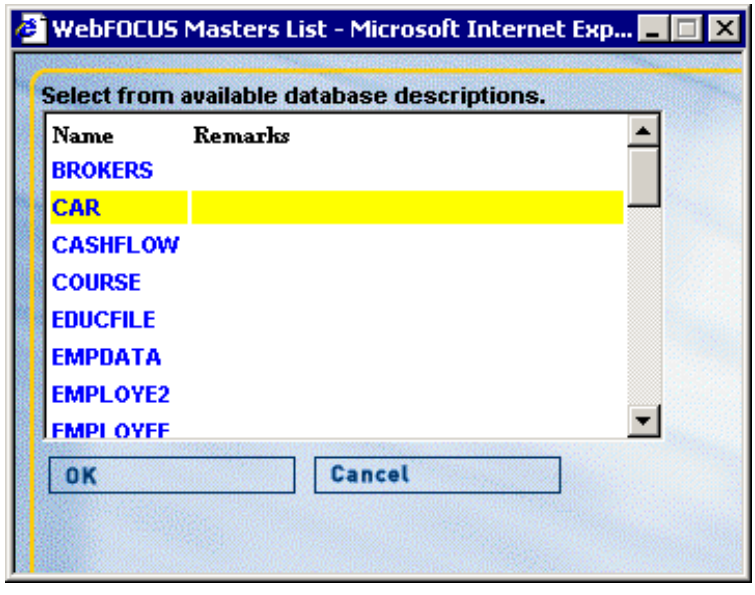
Procedure: How to Create a Custom Report

1. Right-click the *Custom Reports* folder and select *New*.



2. Click the tool with which you want to create the report. The Master File dialog box appears. If you choose New Folder, you can specify a subfolder name and then create your Custom Report using Report Assistant, Graph Assistant, or the Editor.

   Note: Custom Reports can only be created and edited with the HTML version of Report or Graph Assistant, even if you have set the flag for the Java Report or Graph Assistant.



   Note: The list of Master Files is determined by the Default Reporting Server's profile settings or, if set, by the Server and Application Path settings on the Domain. You do not have the ability to set a Server and/or Application Path at the Custom Report level at this time.

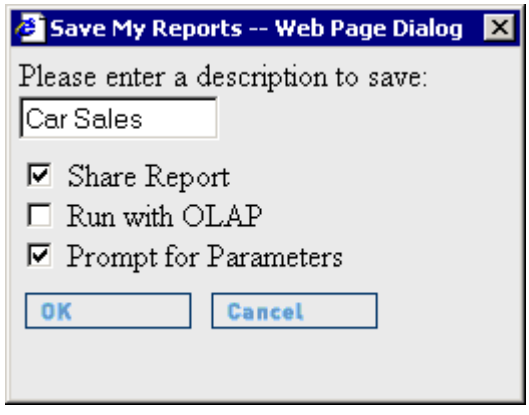
3. Select your Master File and click *OK* to open the tool of your choice.

   Note: You can scroll through the Master File list quickly by typing the first letter of a data source. For example, if you type 'C' the list will advance to the CAR data source.

4. Design your Custom Report.

5. Click *Save As* to open the Save My Reports window.

6. Enter a name for your report and select the options of your choice. For example, check the Share Report check box to share your report with another user.



7. Click *OK* to save your report as a Custom Report.

8. You can access your Custom Report under the Custom Reports folder in the My Reports tab.

9. If you share your custom reports, they will be available to other users using the Shared Reports tab as shown below:

For information about running a Custom Report, see *How to Run a My Report or Custom Report* in the *Using Java Applet Managed Reporting* chapter in the *Managed Reporting End User's Manual*.

In Release 5.2.3, Custom Reports are available in Dashboard. See *Custom Reports in Dashboard* on page 3-19.

Execution of a Custom Report Using -INCLUDE

In Release 5.2.3, you can insert one procedure within another using the -INCLUDE command. When you name a report, Managed Reporting creates the actual file name in the format of username:userfolder/fexname, which tells Managed Reporting where to find the file. The file name is displayed in the Properties dialog box and is the value to specify when referencing the report with -INCLUDE within Custom Reports. Note that only Analytical Users who have been granted the Advanced capability, or Administrators and Developers, who have the Advanced capability by default, can create Custom Reports.

The actual file name for Shared Custom Reports is in the format of shared user's name:shared user's folder/shared procedure name, also designated on a Shared Custom Reports Properties dialog box.

Standard Reports include the procedure name, also designated on a Standard Report's Properties dialog box.

The -INCLUDE command is fully documented in *Execution of a Custom Report Using -INCLUDE* in the *Using Java Applet Managed Reporting* chapter in the *WebFOCUS Managed Reporting End User's Manual*.

Deferred Receipt

You can connect from a single WebFOCUS Client installation to multiple WebFOCUS Reporting Servers. This means that it is possible for a single user to have deferred tickets for output residing on multiple servers. These servers can be on different platforms and may require different user IDs. Users have access to all their deferred output, regardless of its location, and are automatically prompted for credentials as needed.

There are administrator settings for managing deferred workload. There can be up to one alternate deferred server per immediate server to separate interactive and deferred processing. You can limit the number of server agents allocated to handling deferred requests, and also the number of deferred requests a given user can process at one time (these features are not available on OS/390®). On OS/390, a global keyword, UNIQUE, restricts simultaneous server connections to one per logon ID, which can be used to manage both deferred and interactive workload when server authentication is used.

Deferred Receipt is fully documented in the *Using the Deferred Report Status Interface* chapter in the *WebFOCUS Managed Reporting End User's Manual*.

Migration Utility

The command line migration utility has been enhanced to support multiple Managed Reporting repositories and migration across platforms (for example, Windows NT/2000 to UNIX) and code pages (for example, UNIX to OS/390 UNIX). The Public user and group required for the Dashboard are also added to the Managed Reporting repository during migration if not found.

The migration utility is fully documented in the *WebFOCUS and ReportCaster Installation and Configuration Manual* for your platform.

Repository Extract Utility

The IBIDumpRepos.jar utility has been replaced by IBIExtractRepos.jar, which includes support for extracting domain content information, in addition to users, groups, and domain descriptions. Master Files are provided to describe these extracts so that administrators and developers can better analyze the content of their domains with WebFOCUS.

The Extract Utility is fully documented in the Managing Reporting *Extract Utility* chapter in the *WebFOCUS Managed Reporting Administrator's Manual*.

Creating Multiple Content Pages

You can create multiple content pages in your Dashboard view. In previous releases, you were limited to only one content page.

Dashboard administrators, as well as authenticated users, can use this feature to increase the amount of space available to display content in the Dashboard and to help organize content.

Procedure: How to Create Content Pages

1. From the Content window, click *Add Page*.
2. Enter the tab name in the Page Description text box.
3. Click *Update*.

   When a page is added, it is added as the last page. You can rearrange the order of the content pages using the Move Left, Move Right, or Set Default buttons. The Set Default button promotes the current page to the first page.

Creating a Group View

You can create a different view of the Dashboard for each user group. These views are known as group views and are based on Managed Reporting repository groups. For each group view, you can select a banner and customize the colors, composition, and content.

Dashboard administrators can use this feature to create different Dashboard views for Managed Reporting repository groups.

Procedure: How to Add a Group View

1. Open the View Builder.
2. Click *Group Views*.
3. Click *Add*. An input form displays at the bottom of the Group Views window.
4. Select a Group Name from the drop-down list.
5. Enter a description for the group view.
6. Click *Submit*. The new view is added to the Group Views list.

Unique Customizations for Public Views

Dashboard administrators can create different customizations for each public view in the Dashboard. This enables you to select a different look, composition, and message of the day for each public view. Previously, you could only have one customization that applied to all public views.

Dashboard customization is fully documented in the *Customizing Dashboard* chapter in the *WebFOCUS Managed Reporting Administrator's Manual*.

Removing the Banner

You can create your Dashboard view without a banner. Previously, if you wanted to remove the banner, you had to edit the profile.prf file manually, and change the size of the banner so it would not be visible in the Dashboard.

Dashboard administrators can use this feature when customizing Dashboard views using the View Builder.

Dashboard users can take advantage of these enhancements when searching for reports, folders, Web pages, or Web addresses from the Dashboard.

Procedure: How to Remove the Banner in Dashboard

1. From the View Builder, click *Composition*.
2. Deselect the *Check to use Banner* check box.
3. Click *Preview*.
4. Click *Save*.

Customizing Content Block Layouts

You have more control over how content blocks display in the Dashboard. Previously, you were limited to one or two columns and you could not control how much space each column would fill in the Dashboard. You can add as many columns as you need and, using a percentage, you can specify the width of each column.

Dashboard administrators, as well as authenticated users, can use these enhancements to help design the layout of content blocks in their Dashboard view.

Content blocks are fully documented in the *Creating a Content Block* chapter in the *WebFOCUS Managed Reporting Administrator's Manual*.

Saving Most Recently Used Reports

Dashboard automatically creates a list of the last ten reports you have viewed. You can access the list from the Recent link in the banner. The Recent link is included for all authenticated users.

You can clear your list of recently used reports with the Remove All button.

Creating a My Favorite Reports List

There is an Add to Favorites option on the pop-up menu that opens when a user clicks a report in a Domain Tree, Role Tree, list block, or folder block.

You can also add reports to the My Favorites list when you are performing a domain search.

Creating Custom Toolbars

How to:

Create a Toolbox

Copy a Toolbox

Create a Toolbar

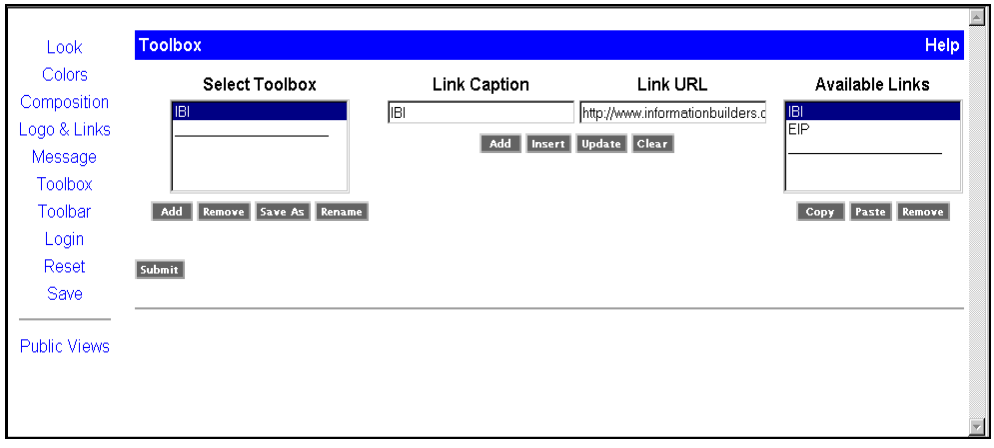
Dashboard Administrators can insert a custom toolbar in the Dashboard that contains any type of link that the Internet or intranet supports, such as Web sites, applications, documents, and other tools. When a user clicks one of the links, a new browser window opens and displays the contents.

Toolbars can be placed in various positions on the Dashboard, and you can select different colors for the toolbars so they conform with your Dashboard view.

Because toolbars are created from items in your toolbox(es), you must create a toolbox before you create a toolbar. You can create as many toolboxes as you require, and each can contain an unlimited number of items.

Procedure: How to Create a Toolbox

1. From the View Builder, click *Toolbox*. The Toolbox window opens:



2. In the Select Toolbox field, click *Add*.
3. Enter a name for the toolbox in the Explorer User Prompt dialog box and click *OK*.
4. In the Link Caption text box, enter the name for the link. This is the text that will display in the toolbar.
5. In the Link URL text box, enter the URL for the link in the format:

   `http://websitename`

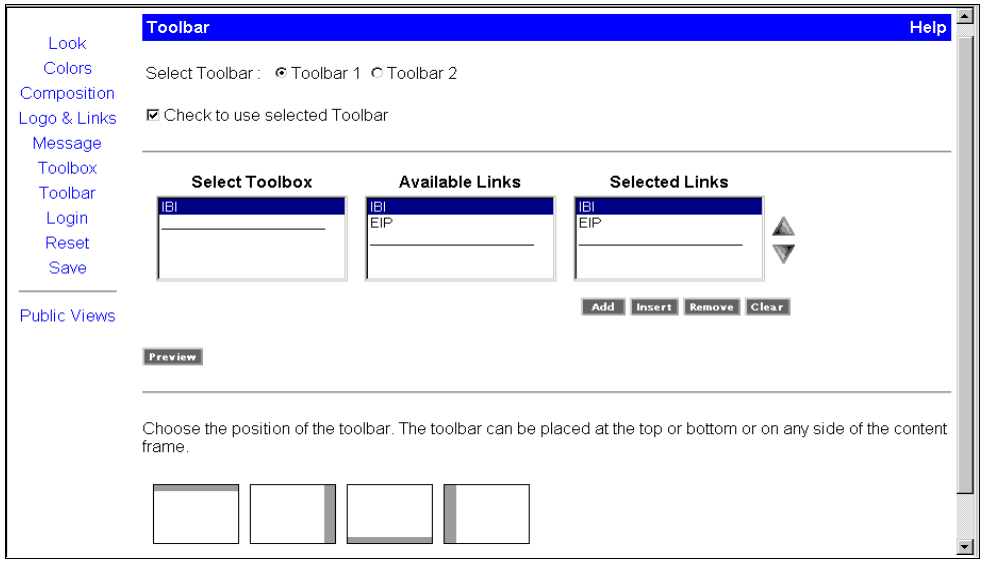
   For example, http://www.informationbuilders.com.
6. Click *Insert*. The link is added to the Available Links box.
7. Repeat steps 4-6 to add additional items to the toolbox. The number of items you can put in a toolbox is unlimited.
8. Click *Submit*.

Procedure: How to Copy a Toolbox

1. From the View Builder, click *Toolbox*. The Toolbox window opens.
2. Highlight a toolbox from the Select Toolbox list and click *Save As*.
3. Enter a name for the new toolbox.
4. Click *OK*.

Procedure: How to Create a Toolbar

1. From the View Builder, click *Toolbar*. The Toolbar window opens.



2. Select the *Toolbar 1* or *Toolbar 2* radio button.
3. Highlight a toolbox in the Select Toolbox list.
4. From the Available Links list, highlight a link and click *Add*.
5. Repeat steps 3 and 4 until your toolbar is complete.
6. Use the up and down arrows to position the order of the selected links.
7. Select the toolbar position (top, bottom, left, or right) by clicking the appropriate image at the bottom of the Toolbar window (you may need to scroll down to view this).
8. Click *Save* to save your changes.

Stand-Alone Domain/Role Trees

To provide more space on your Dashboard view while still allowing users access to the Domain/Role Tree, you can insert a stand-alone Domain/Role Tree in your Dashboard view. The stand-alone Domain/Role Tree can be accessed from a link in the banner named Tree.

Dashboard administrators can use this feature when they want to provide users with access to the Domain/Role Tree, but not have it display in the Dashboard view.

Positioning Domain and Role Trees is fully documented in *Positioning the Domain Tree, Role Tree, and Banner* in the *Customizing Dashboard* chapter in the *WebFOCUS Managed Reporting Administrator's Manual*.

Optionally Displaying Frame Separators

You can choose to keep or omit frame separators from the Dashboard view. Omitting frame separators creates a seamless look between the banner and the sidebar (Domain/Role Tree).

Frame separators are fully documented in *Positioning the Domain Tree, Role Tree, and Banner* in the *Customizing Dashboard* chapter in the *WebFOCUS Managed Reporting Administrator's Manual*.

Hiding the Display of Content Blocks

You can hide the display of a content block from the Dashboard view by selecting the deactivate option when adding or editing a content block.

This is useful when you want to temporarily remove a content block from a Dashboard view.

Procedure: How to Temporarily Remove a Content Block

1. From the Content window, select the content page where the content block is located.
2. Click the check box next to the block you wish to edit.
3. Click *Edit Block*. The Edit Block window opens.
4. Select the *Check to Inactivate Block* box.
5. Click *Save*.

Default Public User

The public user is automatically created in Managed Reporting, so you do not need to create one for new installations of the Dashboard. Prior to Release 5.2, you were required to create a public user in Managed Reporting before customizing your Dashboard view.

Domain Searches

When performing a domain search in the Dashboard, you can search across domains based on modification dates for a file. For example, you can search for documents that were modified between certain dates, or since a specified date.

The format of the date and time shown in the Domain Search Results window and the Modified Since panel are based on your system setting.

User Options Settings

There is a link in the Dashboard banner called Options that allows you to set certain user options. Some of these include:

- Selecting the tool type (HTML or Java applet) for Report Assistant and Graph Assistant.
- Controlling report execution windows.
- Selecting the maximum number of recently used reports.
- Selecting how menus are displayed.
- Selecting how folders in a domain are displayed.

User options are fully documented in *Setting User Options* in the *Using Dashboard* chapter in the *WebFOCUS Managed Reporting End User's Manual*.

Custom Reports in Dashboard

In Release 5.2.3, custom reports enable you to create your own reports using the Report Assistant, Graph Assistant, or text editor. Depending on which tool you use to create your report, you can edit your report using the same tool. You can change the name of your custom report from the Properties window.

Procedure: How to Create a Custom Report in Dashboard

1. In the Domain Tree, expand the My Reports folder.
2. Click the Custom Reports folder and select one of the following:
   - *Report Assistant* to create a report using the Report Assistant.
   - *Graph Assistant* to create a graph using the Graph Assistant.
   - *Editor* to create a report or graph using the text editor.
   - *New Folder* to create a new folder under Custom Reports.

   If you have selected the Report Assistant or Graph Assistant, you will be prompted to select a data source from which you want to report. Click *OK* to continue.

   The appropriate tool opens.
3. Design and then save your Custom Report.

Filters For Standard Reports in Dashboard

In Release 5.2.3, users can select predefined filters when running Standard Reports in Dashboard. Administrators define the filters when they build the reports.

Procedure: How to Use Filters

1. Click the Standard Report.
2. From the menu, select *Filter*.

   If the Filter option is not available on the menu, this means filters have not been set up for this report.

   The Filter Selection window opens.
3. Select the options you want to filter on and click *Save*.

   Your selections are saved and the Filter Selection window closes.
4. Click the report and select *Run* or *Run Deferred* to execute the report.

Content Pages: ReportCaster and Report Library

In Release 5.2.3, administrators and authenticated users can add content pages where users can view the ReportCaster and the Report Library user interfaces. Add these pages where you create all other content pages in the View Builder or from the Personalize link in the Dashboard.

Adding the ReportCaster and Report Library interfaces to a content page is different from accessing them from banner links, in that a new browser window does not open. Each interface appears in its own content page right in the Dashboard.

Since public users do not have access to ReportCaster or the Report Library, administrators will not be able to create these types of pages for public users.

ReportCaster and Report Library content pages are fully documented in *Creating Content Pages* in the *Creating a Content Block* chapter in the *WebFOCUS Managed Reporting End User's Manual* and the *WebFOCUS Managed Reporting Administrator's Manual*.

Dynamic Tracing

In Release 5.2.3, you can turn on tracing dynamically in the Dashboard by editing the bid-config.xml file in the worp\conf\ directory. In the bid-config.xml file, find the following line

```
<trace-option trace-flag="false" trace-level="DEBUG"/>
```

and set the value to true.

You do not need to reload the servlet.

CHAPTER 4

Ad Hoc Reporting Enhancements

This chapter describes all of the features for ad hoc reporting in Version 5 Release 2 and has been updated to reflect Version 5 Release 2.3. Topics that contain Version 5 Release 2.3 features are marked with asterisks.

Topics:

OLAP

* HTML Report Assistant

* HTML Graph Assistant

* HTML Report Assistant and HTML Graph Assistant

OLAP

OLAP has been rewritten using JavaScript™ and HTML instead of Java applets. OLAP's functionality, look, and feel remain the same. OLAP enhancements include the following:

- Drag-and-Drop Dimensions and Measures.
- Right-Click in a Report.
- Hidden Reports.
- Tiles.

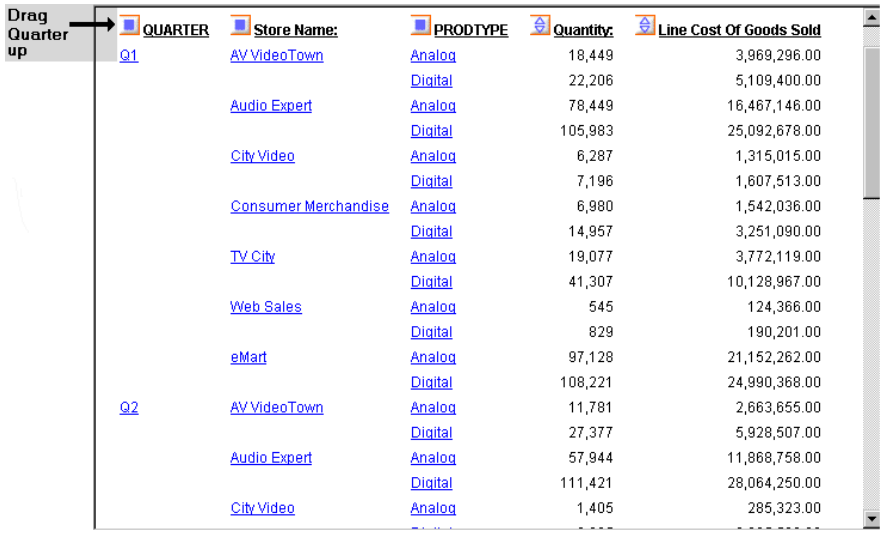
OLAP is fully documented in the *Analyzing Data in an OLAP Report* chapter in the *WebFOCUS Managed Reporting End User's Manual*.

Drag-and-Drop Dimensions and Measures

You can drag and drop measures and dimension controls in an OLAP-enabled report from one position to another. These drill-down and drill-across positions are as follows: By to By, By to Across, Across to By, and Across to Across. You can also drag and drop dimension controls from above the blue-colored band to the body of the report. You can drag and drop measures to other measure positions to affect the order in which you read the measures.

Example: Using the Drag-and-Drop Feature in a WebFOCUS OLAP Analysis

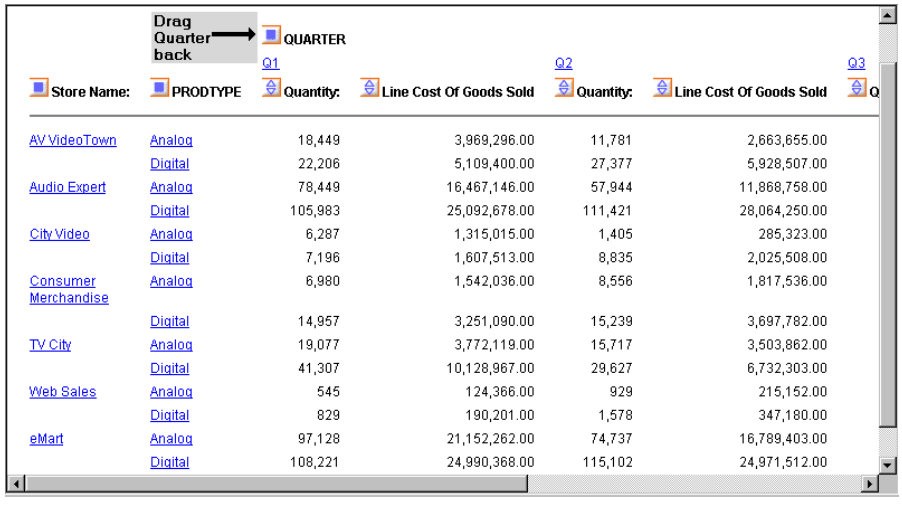
In the following report, quarterly production information is spread out. A horizontal display may make comparisons easier.



QUARTER	Store Name:	PRODTYPE	Quantity:	Line Cost Of Goods Sold
Q1	AV VideoTown	Analog	18,449	3,969,296.00
		Digital	22,206	5,109,400.00
	Audio Expert	Analog	78,449	16,467,146.00
		Digital	105,983	25,092,678.00
	City Video	Analog	6,287	1,315,015.00
		Digital	7,196	1,607,513.00
	Consumer Merchandise	Analog	6,980	1,542,036.00
		Digital	14,957	3,251,090.00
	TV City	Analog	19,077	3,772,119.00
		Digital	41,307	10,128,967.00
	Web Sales	Analog	545	124,366.00
		Digital	829	190,201.00
	eMart	Analog	97,128	21,152,262.00
		Digital	108,221	24,990,368.00
Q2	AV VideoTown	Analog	11,781	2,663,655.00
		Digital	27,377	5,928,507.00
	Audio Expert	Analog	57,944	11,868,758.00
		Digital	111,421	28,064,250.00
	City Video	Analog	1,405	285,323.00

1. Drag and drop *QUARTER* above the report.

   The report changes immediately and appears as follows. The store information is more compact.

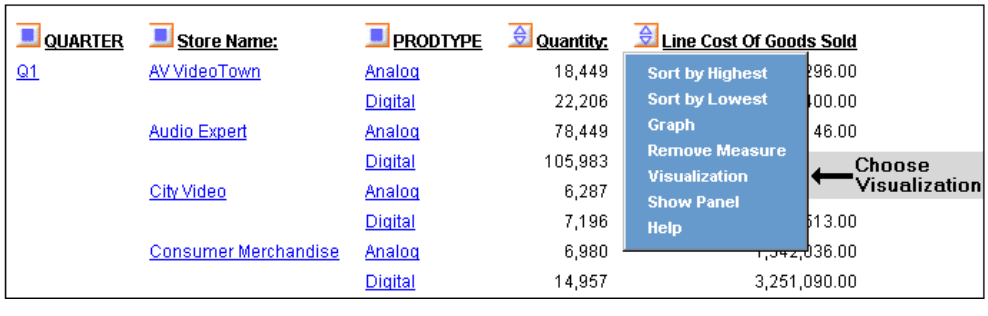


Right-Click in a Report

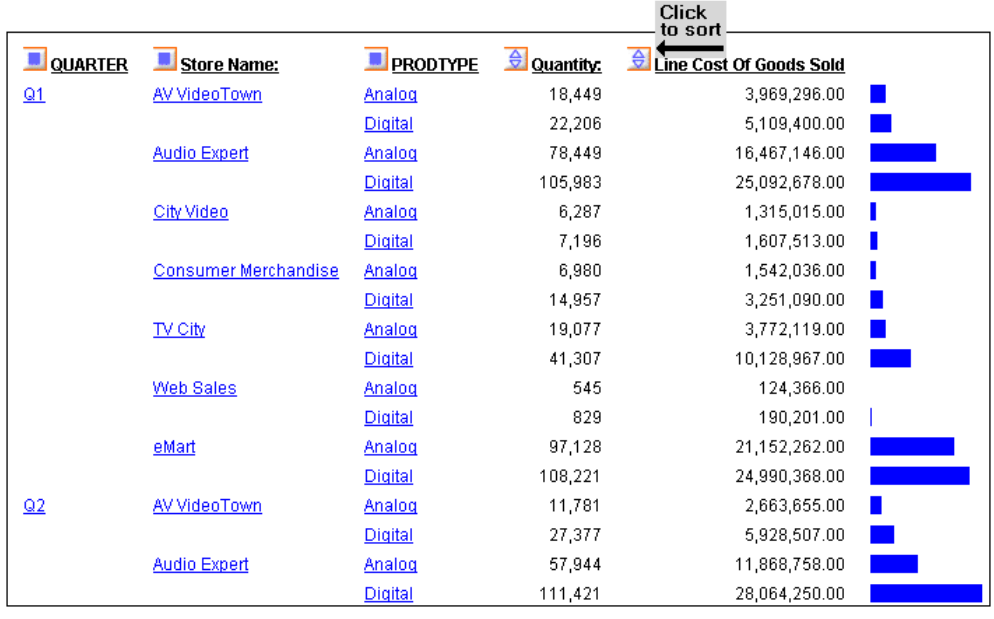
You can right-click dimensions and measures in an OLAP-enabled report to yield several choices. When you right-click a dimension, you have the following choices: Delete, New, Move to Across/By, Full Screen/Show Panel, Field Info., and Help. When you right-click measures you have the following choices: Sort by Highest, Sort by Lowest, Graph, New, Remove Measure, Remove Visualization, Forecast..., Full Screen/Show Panel, Field Info., and Help.

Example: Using the Right-Click Feature in a WebFOCUS OLAP Analysis

1. Right-click *Line Cost of Goods Sold* and choose *Visualization*. This applies a data visualization bar graph to each value in the column, which may reveal a trend.



2. Once again, the display changes instantly, as shown here.

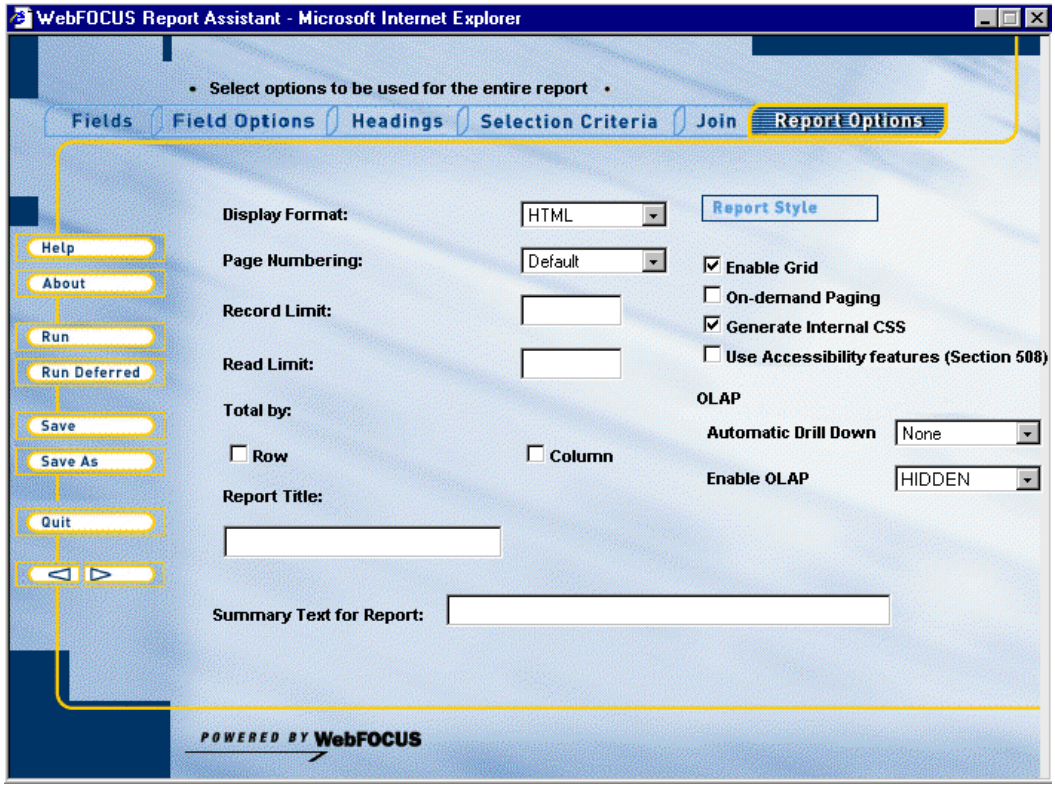


QUARTER	Store Name:	PRODTYPE	Quantity:	Line Cost Of Goods Sold
Q1	AV VideoTown	Analog	18,449	3,969,296.00
		Digital	22,206	5,109,400.00
	Audio Expert	Analog	78,449	16,467,146.00
		Digital	105,983	25,092,678.00
	City Video	Analog	6,287	1,315,015.00
		Digital	7,196	1,607,513.00
	Consumer Merchandise	Analog	6,980	1,542,036.00
		Digital	14,957	3,251,090.00
	TV City	Analog	19,077	3,772,119.00
		Digital	41,307	10,128,967.00
	Web Sales	Analog	545	124,366.00
		Digital	829	190,201.00
	eMart	Analog	97,128	21,152,262.00
		Digital	108,221	24,990,368.00
Q2	AV VideoTown	Analog	11,781	2,663,655.00
		Digital	27,377	5,928,507.00
	Audio Expert	Analog	57,944	11,868,758.00
		Digital	111,421	28,064,250.00

Hidden Reports

You can turn the OLAP selections panel on but keep it hidden in an OLAP-enabled report. Later, if you want to view the OLAP selections panel, you can then right-click any dimension or measure and select Show Panel. You can also return to the Report Options tab in Report Assistant and select Top or Bottom under Enable OLAP to turn on the selections panel.

In WebFOCUS, OLAP options are available on the Report Options tab of the Report Assistant. The relevant option, Enable OLAP, is located in the lower right corner.



Tiles

You can group numeric data into any number of tiles (percentiles, deciles, quartiles, and so on) in tabular reports. For example, you can group students' test scores into deciles to determine which students are in the top ten percent of the class or determine which salesmen are in the top half of all salesmen based on total sales.

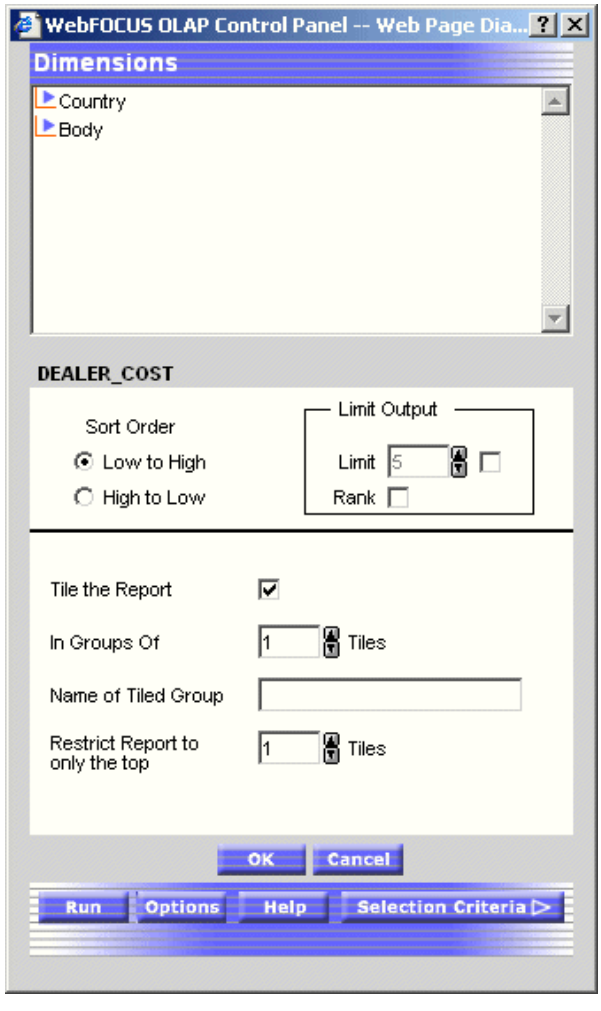
Grouping is based on the values in the selected vertical (BY) field, and data is apportioned as equally as possible into the number of tile groups you specify.

When you group data into tiles:

- A new column (labeled TILE by default) is added to the report output and displays the tile number assigned to each instance of the tile field. You can change the column title in the Tiles section of the OLAP Control Panel.
- Tiling is calculated within all of the higher-level sort fields in the request, and restarts whenever a sort field at a higher level than the tile field's value changes.
- Instances are counted using the tile field. If the request displays fields from lower-level segments, there may be multiple report lines that correspond to one instance of the tile field.

Procedure: How to Group Data Into Tiles in an OLAP Report

1. Open the OLAP Control Panel.
2. Select a numeric or date field from the Drill Down box.
3. Click the *Tiles* button. The Tiles pane opens.



4. Select the *Tile the Report* check box.
5. In the *In Groups Of* spin box, select the number of tiles to be used in grouping the data. For example, 100 tiles produces percentiles, 10 tiles produces deciles, and so on.
6. In the *Name of Tiled Group* input box, type a name for the Tile column.

7. In the *Restrict Report to only the top* spin box, select the number of tile groups to display in the report.
8. Optionally, select a Sort Order option button:
   - Choose *High to Low* to sort data in descending order so that the highest data values are placed in tile 1.
   - Choose *Low to High* to sort data in ascending order so that the lowest data values are placed in tile 1. Low to High is the default value.
9. If you wish to specify the highest tile value to appear in the report, select a value from the Limit spin box. For example, if you enter a Limit of 3, the report will not display any data row that is assigned a tile number greater than 3.
10. Click *OK* to accept the selections and return to the main Control Panel window.
11. Click *Run* to reexecute and view the report.

HTML Report Assistant

HTML Report Assistant enhancements include the following:

- Report Presentation.
- Creating New Columns.
- Selecting Records for a Report.
- Designing Joins.
- End User Drill Down Capability.
- Performing OLAP Operations Directly From the Report.

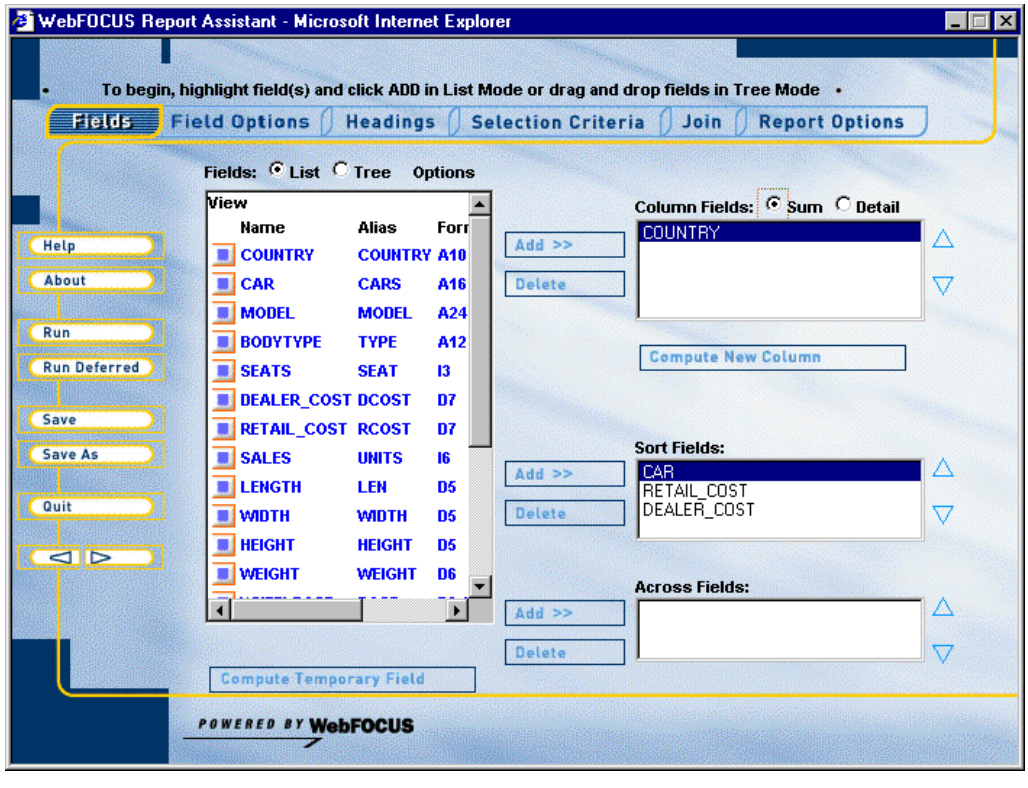
HTML Report Assistant is fully documented in the *Creating a Report With Report Assistant* chapter in the *WebFOCUS Managed Reporting End User's Manual*.

Report Presentation

The HTML Report Assistant is an HTML-based reporting tool that allows you to select a data source, specify any sorting or grouping information, and display the report in your browser or another desktop application.

You can access the HTML Report Assistant through the Dashboard, Managed Reporting, or as a stand-alone version.

The HTML Report Assistant is divided into the following tabs—Fields, Fields Options, Headings, Selection Criteria, Join, and Report Options. Each tab includes explanatory text designed to guide you in using the specific features accessed within the tab. Availability of tabs and buttons depends on the type of user logging on to WebFOCUS or Dashboard. Self-service users do not have access to the Join tab, Run Deferred, Save, and Save As buttons or OLAP options in the Report Options tab.



WebFOCUS lists available fields in two formats—as a list of fields or as a tree separated by segments. You can select the Fields radio button to toggle between the two formats. Select report fields by highlighting the field and clicking *Add* for Column Fields, Sort Fields, and Across Fields. From either the List or Tree view, you can drag and drop or use *Add*. To multi-select, use List to include fields and then drag and drop or click *Add*. List is the default value.

List mode is the default for the Field List Box. List mode allows you to view one or more options—name, alias, title, remarks, format, description, segment, colno, and file name. In List mode, you can sort any list of variables alphabetically from A to Z or from Z to A by clicking the option name (for example, name or alias). In tree mode, the following options are provided in a box under the tree: segment, alias, title, description, and format.

When you specify a column, you must designate the selected field as Sum (WebFOCUS adds the values together) or Detail (WebFOCUS prints the values individually). Sum is the default selection.

In Report Options, you can enter your own report title and summary text description for your report. Also, the TABT, EXL97, and EXL2K FORMULA have been added to the display format drop-down list.

Creating New Columns

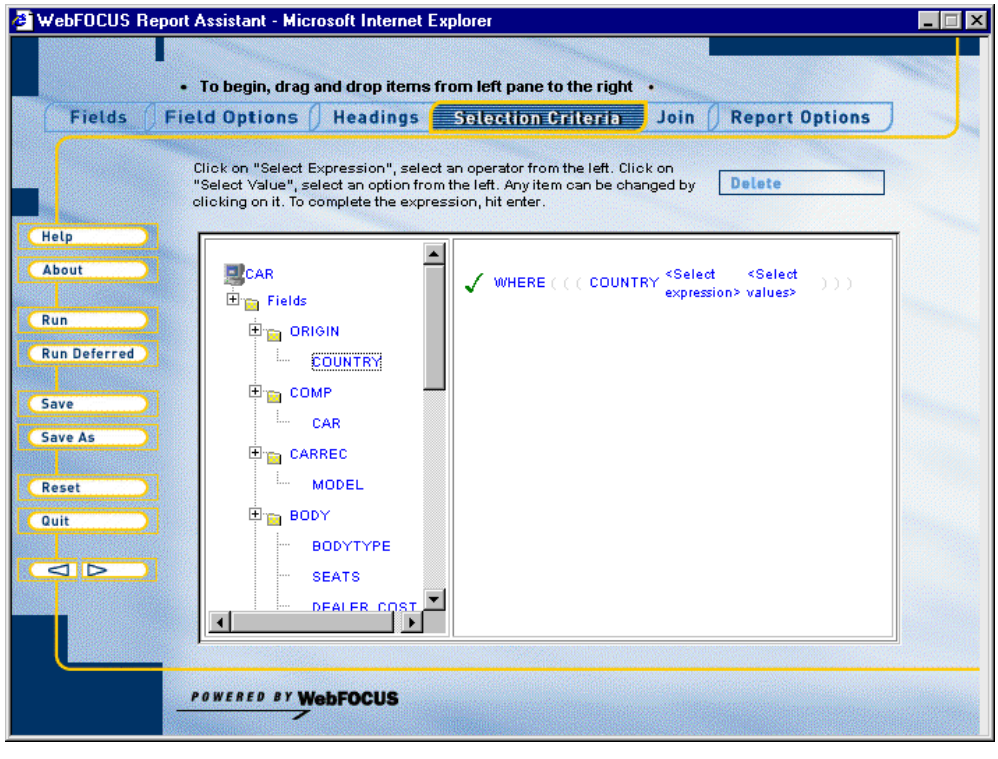
You can use the Fields tab to compute new columns (COMPUTE) and virtual fields (DEFINE). If you attempt to create a new column with more than 66 characters, the following message appears: "Compute string cannot be more than 66 characters." If you attempt to create a new virtual field with greater than 66 characters, the following message appears: "Define string cannot be greater than 66 characters."

Selecting Records for a Report

The Selection Criteria tab enables you to create WHERE or WHERE TOTAL statements for record selection based on values, parameters, and fields. You can combine expressions by clicking the default operator AND and switching between the choices AND, OR, WHERE, and WHERE TOTAL.

Procedure: How to Select Records Based on Values, Parameters, and Fields

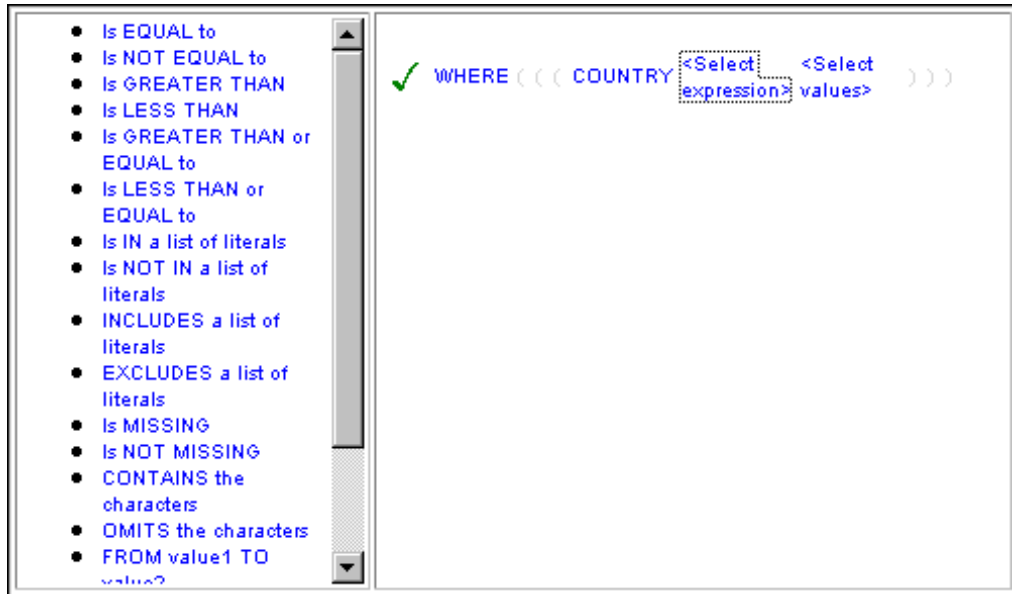
1. Select a field from the Fields tree and drag the field to the right frame. The field name and <Select expression> and <Select values> will appear in the right frame, prefixed by a green check mark.



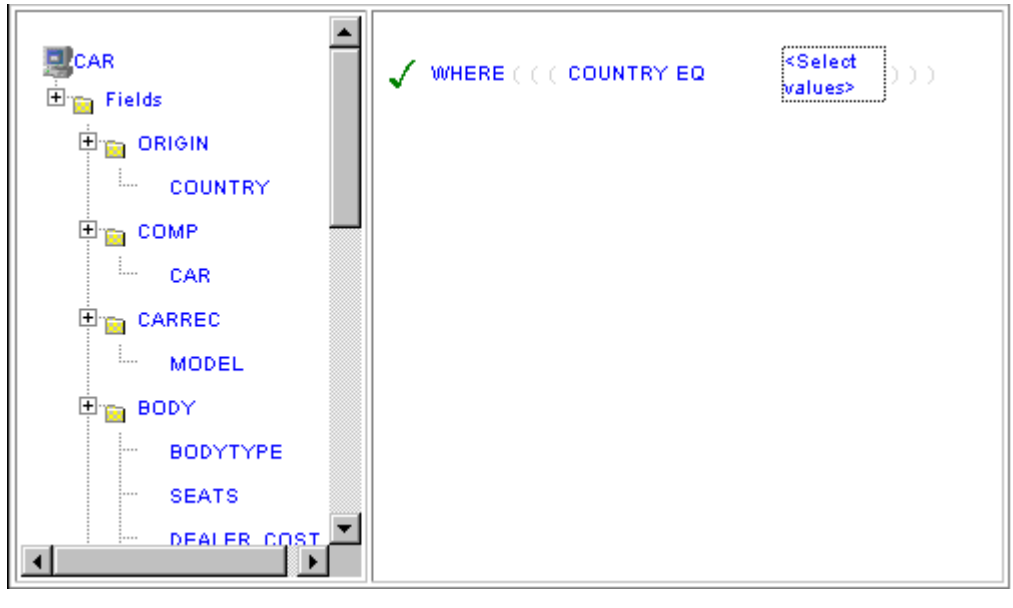
Note:

- You can replace the field you selected by clicking on the field in the expression and then double-clicking the new field in the tree. Your new selection automatically appears in the expression.
- The Fields list will be in tree format. If filters exist, they will be listed prior to the tree listing of the file.

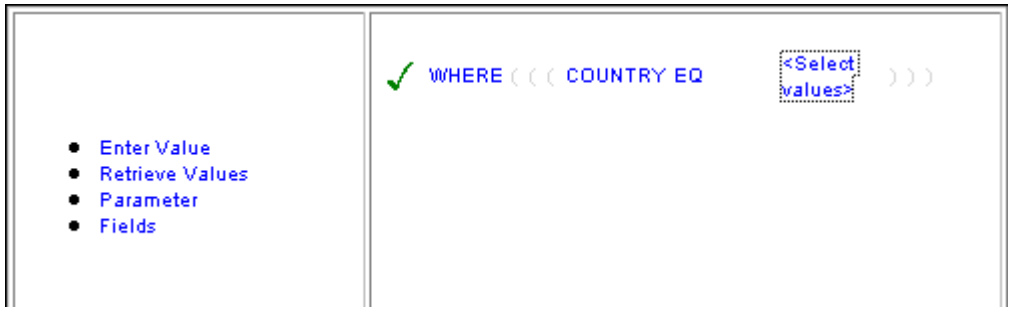
2. Click *<Select expression>* in the right frame to display a list of operators in the left frame:



3. Click the expression of your choice.

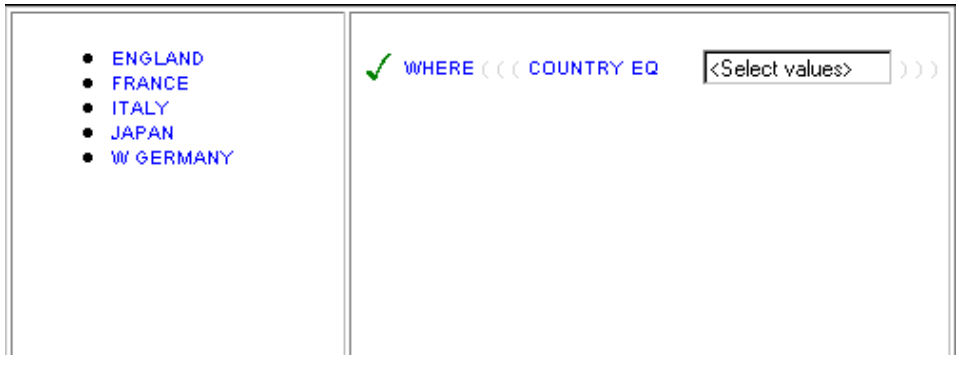


4. Click <*Select values*> to complete the expression.



5. Compare type options will appear in the left frame:

- **Enter or Retrieve Value.** Select to generate a list of values for the field and display an edit box to type in the value.

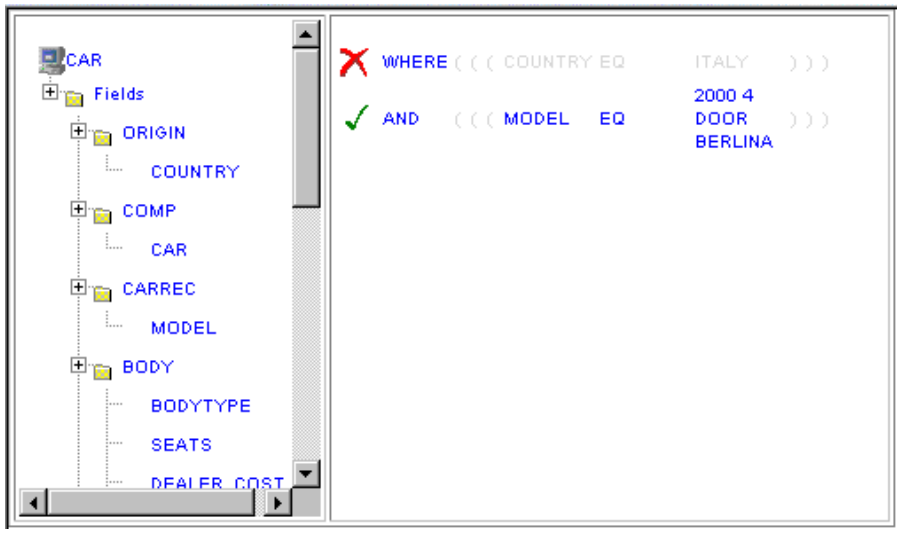


Note: When entering a value that contains a literal, enclose the value in single quotation marks. When entering multiple values, they must be delimited with "or".

- **Parameter.** Select to display an Edit box to enter the name of the &variable. Click *Add* to complete the expression.

- **Field.** Select to regenerate the fields list.

You can activate or delete expressions by switching between the green check and the red X that appear next to each expression.You can also use parentheses and group expressions together to optimize the WHERE statement. Click the grayed-out parentheses to activate both.



Filters are listed above the file in the fields frame of the Selection Criteria tab. You can drag and drop the filter of your choice from the fields frame to the expressions frame.

Designing Joins

HTML Report Assistant has a redesigned Join Tool which you access from the Join tab. The process of creating joins involves selecting each Master File you want to add and then selecting your joins. When you click the Create Join button, the join name and syntax are displayed. You can also edit or delete a join from the Existing Join statements list. See *Joins* in the *Creating a Report With Report Assistant* chapter in the *Managed Reporting End User's Manual*.

End User Drill Down Capability

Along with Managed Reporting Administrators and Developers, end users can drill down to Standard Reports from within Reporting Objects and My Reports. In Release 5.2.3, end users can also drill down from one My Report to another My Report. The end user drill down capability enables users to:

- Create drill downs to Standard Reports from Reporting Objects.
- Run or delete drill downs created by Administrators or Developers to Standard Reports from My Reports. Delete does not alter the Reporting Object, since it only applies to ad hoc requests.

Note: The default is that functionality exists for all users.

Performing OLAP Operations Directly From the Report

In Release 5.2.3, the Enable OLAP options in the Report Assistant and the OLAP Panel options in Developer Studio control how users can interact with an OLAP report and access OLAP tools. For Standard Reports delivered to Managed Reporting users, the Managed Reporting content developers make these decisions. However, users who are creating their own reports can OLAP-enable them and control the OLAP interfaces and drill down options.

The NONE option turns off the OLAP Control Panel and the OLAP Selections pane, but allows OLAP functionality from the report itself. You can access options on right-click menus, drag and drop columns within the report, and use up and down arrows to sort columns from high to low or vice versa. NONE is the default value.

HTML Graph Assistant

In this section:

Addition of Several Templates to the Styles Tab

Updated Styles Tab to Include Graph Types and Graph Styles

Addition of the Preview Screen

Auto Update Feature Added to the Preview Screen

The HTML Graph Assistant is an HTML-based graphical tool that enables you to create a graphical representation of your data. The HTML Graph Assistant is accessible through the Dashboard and Managed Reporting.

The HTML Graph Assistant is divided into the following tabs:

- **Styles.** Select a graph type. Graph types include line graphs (connected point plots), bar graphs, pie graphs, and scatter graphs, as well as variations on these types.
- **Fields.** Add and remove data fields from your graph.
- **Field Options.** Add conditional styling, drill down, and other options to your graph.
- **Across.** Create multiple graphs by adding a second X-axis.
- **Headings.** Add style headings and/or footings for your graph, as well as horizontal (X) and vertical (Y) axis labels.
- **Selection Criteria.** Limit the data that appears in your graph by creating WHERE statements. WHERE statements limit data by creating parameters the data must satisfy before it is included in the data set.
- **Properties.** Customize your graph using the subtabs (Options, Settings, X-axis, Y-axis, Pie) of the Properties tab. These include customizing fonts, X-axis and Y-axis orientation, output destination, graphs on servers, label locations for all graph types, legend location, and much more.

HTML Graph Assistant is fully documented in the *Creating a Graph With Graph Assistant* chapter in the *WebFOCUS Managed Reporting End User's Manual*.

HTML Graph Assistant enhancements include:

- Addition of Several Templates to the Styles Tab.
- Updated Styles Tab to Include Graph Types and Graph Styles.
- Addition of the Preview Screen.
- Auto Update Feature Added to the Preview Screen.

HTML Graph Assistant is fully documented in the *Creating a Graph With Graph Assistant* chapter in the *WebFOCUS Managed Reporting End User's Manual*.

Addition of Several Templates to the Styles Tab

In Release 5.2.3, the following graph types are available from the Graph Types drop-down list in the Styles tab: line graphs, bar graphs, pie graphs, scatter graphs, area graphs, 3D graphs, bubble charts, polar graphs, radar graphs, spectral map charts, stock charts, and histograms. When you select a graph type, a thumbnail image and brief description of the graph are displayed.

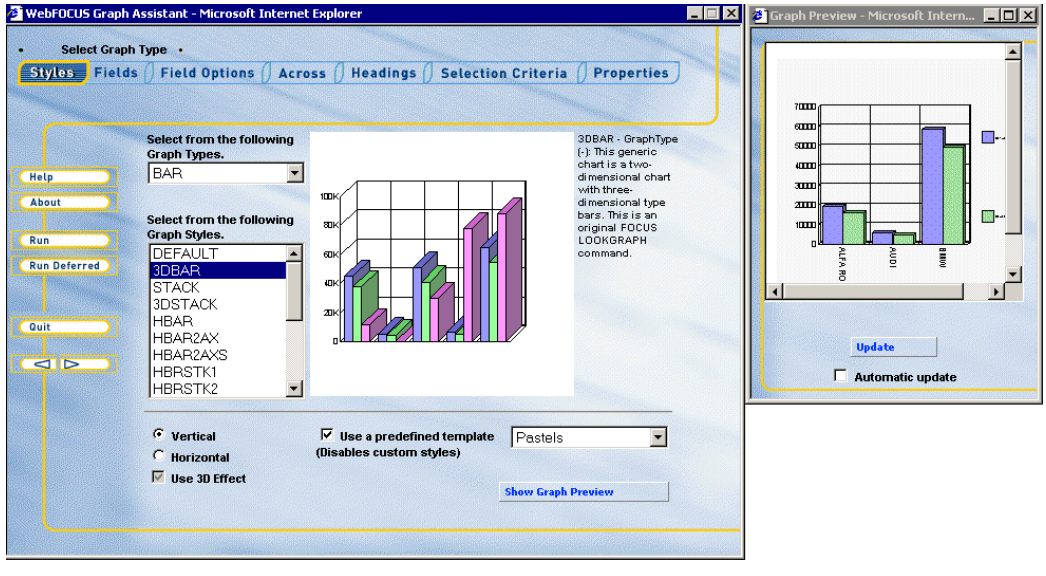
Updated Styles Tab to Include Graph Types and Graph Styles

In Release 5.2.3, you can select from a number of basic graph types, as well as refinements on these types, known as graph styles. The Styles tab provides a list and brief descriptions of the many graph types and styles available in WebFOCUS.

Addition of the Preview Screen

In Release 5.2.3, the graph in the Graph Preview window displays the graph type and style you chose as well as any styling information you specified from the template drop-down list or in the Properties tab. The Preview is best viewed using 1024 x 768 resolution. The Preview does not open by default for other resolutions. In these cases, you have to click the Show Preview button.

The following bar graph uses the 3DBAR graph style. This generic chart is a two-dimensional chart with three-dimensional type bars.



The data used for this Graph Preview is built with the selected data sources' field format types and lengths of the data. In this case, the X-axis field (COUNTRY) is alphanumeric (A10) and will be displayed as Axxxxxxxxxx as the series label

where:

A

Represents the data type alphanumeric.

xxxxxxxxxx

Indicates the number of characters or digits allowed.

The Y-axis (dealer cost and retail cost) scale is based on a data range of selected Y-axis fields.

Auto Update Feature Added to the Preview Screen

In Release 5.2.3, you can update the Graph Preview manually by clicking the Update button to see how your changes will look in a graph. You can also choose to have the Graph Preview updated automatically any time you make changes by selecting the Automatic update check box.

HTML Report Assistant and HTML Graph Assistant

Enhancements to both the HTML Report Assistant and the HTML Graph Assistant include:

- DBA and DBAFILE Support.
- Creating a WHERE Statement Using Date and Time Values.

DBA and DBAFILE Support

WebFOCUS tools, including Report Assistant, Graph Assistant, and JOIN tool components, support DBA files and structures containing DBA.

DBA rules, syntax, and examples are fully described in the *Providing Data Source Security:DBA* chapter in the *Describing Data With WebFOCUS Language* manual.

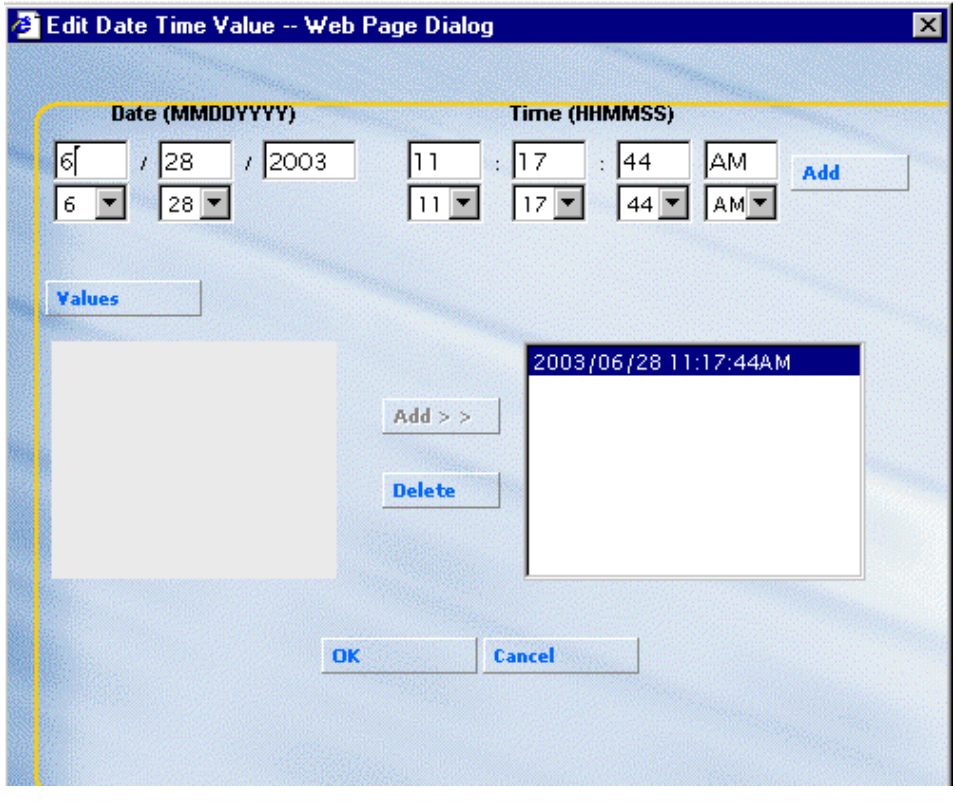
DBA security for joined data structures, is fully documented in the *Joining Data Sources* chapter in the *Creating Reports With WebFOCUS Language* manual.

Creating a WHERE Statement Using Date and Time Values

In Release 5.2.3, you can use the new Date and Time dialog box to enter or choose values for the Date (month, day, and year) and Time (hours, minutes, seconds).

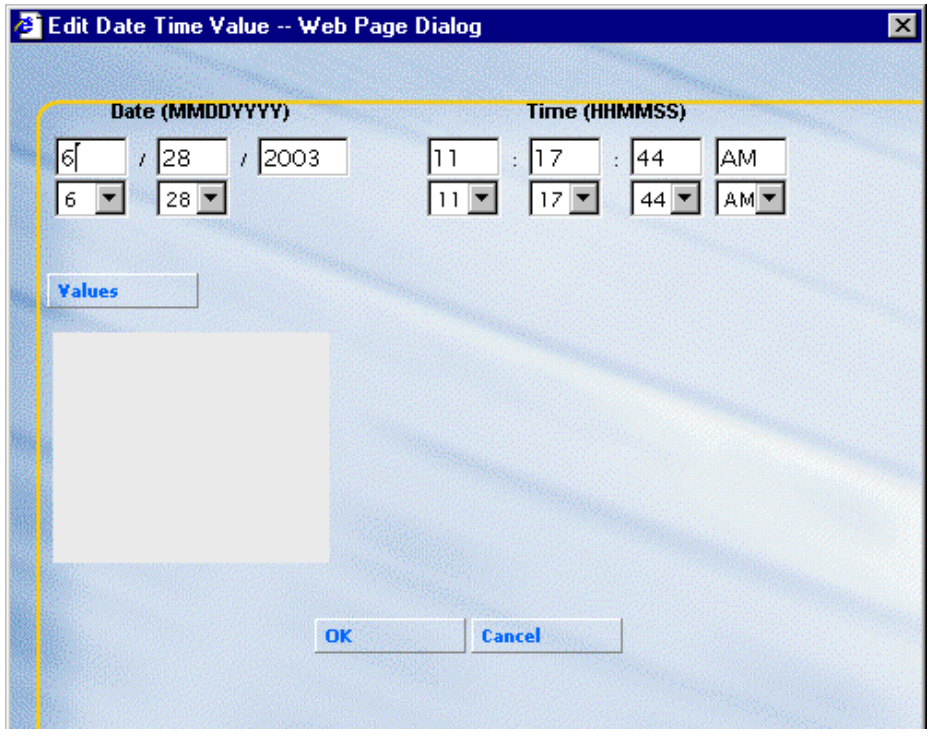
Procedure: How to Create a Where Statement Using Date and Time Values

1. Select a field from the Fields tree and drag the field to the right frame. The field name and <Select expression> and <Select values> will appear in the right frame, prefixed by a green check mark.
2. Click <*Select expression*> in the right frame to display a list of operators in the left frame. Click the expression of your choice.
3. Click <*Select values*> to complete the expression.
4. Click *Enter Value* or *Retrieve Values* to access the Edit Date Time Value window.



5. Type in or use the drop-down boxes to create a new date. Click *Values* to retrieve actual values from the database.

6. If the where operator allows for only one value, for example, greater than, the following window appears.



CHAPTER 5

ReportCaster Enhancements

This chapter describes all of the features for ReportCaster in Version 5 Release 2 and has been updated to reflect Version 5 Release 2.3. Topics that contain Version 5 Release 2.3 features are marked are marked with asterisks.

The following topics describe new features for ReportCaster, ReportCaster API, and Two-Way Email API. For Release 5.2, ReportCaster has been greatly enhanced and expanded to include many new features. In order to familiarize yourself with the new ReportCaster interfaces and features, Information Builders recommends reading the following ReportCaster manuals along with this chapter:

- *ReportCaster Development and Administration*
- *ReportCaster End User's Manual*
- *ReportCaster and Two-Way Email for Self-Service Applications*
- *WebFOCUS and ReportCaster Installation and Configuration* manual for your platform

Topics:

Architecture

* Migration Tools

* Configuring the ReportCaster Web Application

* Configuration Tools

* Security Configurations

* ReportCaster Authentication Exit

ReportCaster User Credentials

* Configuring Access to the User Interfaces

Development and Administration Interface

ReportCaster Server Configuration Tool

Creating and Managing ReportCaster User IDs

* Distribution Enhancements

* Scheduling Wizard and HTML User Interface

Report Library

* ReportCaster Console

Log Enhancements

Configuring ReportCaster for NLS

* ReportCaster API

Two-Way Email API

Architecture

ReportCaster distribution has been moved from the WebFOCUS Reporting Server to the ReportCaster Distribution Server. In addition, ReportCaster supports multiple WebFOCUS Reporting Servers to create and run schedules. ReportCaster communication from the ReportCaster Distribution Server to the WebFOCUS Client supports only WebFOCUS Servlet configurations. WebFOCUS CGI and ISAPI configurations are not supported. For details about how ReportCaster processing works, see *ReportCaster Processing* in the *Introducing ReportCaster* chapter in the *ReportCaster Development and Administration* manual.

Enhancements to the ReportCaster Repository have been made to account for the expanded architecture and the increased functionality of ReportCaster (for example, you can now specify text within the body of an e-mail). For detailed information about the repository tables, see *ReportCaster Repository Tables* in the *ReportCaster Repository Reports and Tables* appendix in the *ReportCaster Development and Administration* manual.

Migration Tools

In Release 5.2.3, Windows users can take advantage of GUI tools for migrating and creating the ReportCaster Repository. These tools are accessible from the Programs menu by selecting *ReportCaster 52* and then *Utilities*. Using these tools, you can easily extract and load data as well as run a Change Management (migrate specific records) by providing your old ReportCaster configuration file (bkrsched.cfg or dserver.xmls). For more information about how to use these migration tools, see the *Migrating ReportCaster Data* chapter in the *WebFOCUS and ReportCaster Installation and Configuration for Windows* manual.

Configuring the ReportCaster Web Application

In Release 5.2.3, you can change the default /rcaster context path used in URLs for accessing ReportCaster. For example, instead of the default URL:

```
http://webserver/rcaster/console/RCHConsole.jsp
```

you can use

```
http://webserver/contextpath/console/RCHConsole.jsp
```

Information on this customization is included throughout the *WebFOCUS and ReportCaster Installation and Configuration* manual for your platform. For WebFOCUS and ReportCaster files, you can either specify this setting during the WebFOCUS and ReportCaster installation, or you can manually change it afterwards. For the Web and Application servers, you must specify your custom context path instead of /rcaster when deploying the rcaster52.war file. Changing the /rcaster context path is not certified on OS/390 and z/OS, and is not supported with Two-Way Email.

Configuration Tools

In Release 5.2.3, using configuration options available on Windows from the ReportCaster52 Configuration menu, you can:

- Enable and disable tracing outside of the ReportCaster Server Configuration tool. From the Programs menu, select *ReportCaster52*, *Configuration*, and then *Traces On* to display the possible tracing options. For detailed information about the ReportCaster tracing options, see the *ReportCaster Tracing* chapter in the *ReportCaster Development and Administration* manual. This option is also available on UNIX and z/OS using a shell script.
- Edit the ReportCaster Server Configuration tool. From the Programs menu, select *ReportCaster52*, *Configuration*, and then *Edit*. For new feature information about the configuration tool, see *ReportCaster Server Configuration Tool* on page 5-5.
- Test the ReportCaster configuration even if the Application Server and ReportCaster Distribution Server are not running. From the Programs menu, select *ReportCaster52*, *Configuration*, and then *Test*. This option is also available on UNIX and z/OS using a shell script.

Security Configurations

ReportCaster supports trusted security configurations and remote authentication. For more information about the various security settings that enable ReportCaster to support this functionality, see *ReportCaster Security Settings* and *ReportCaster Remote Authentication* in the *ReportCaster Security* chapter in the *ReportCaster Development and Administration* manual.

To implement a single signon environment where the Web server identifies users, you can use Managed Reporting signon integration with Basic Web server authentication. In Release 5.2.3, ReportCaster is fully supported when using this signon integration technique. For more information about how to implement this signon integration technique, follow the steps outlined in the *Integrating WebFOCUS Security With Basic Web Server Authentication* chapter in the *WebFOCUS Security and Administration* manual.

ReportCaster Authentication Exit

You can use a user-written, external Java program with ReportCaster, ensuring that ReportCaster user IDs and passwords are properly authenticated without having to store passwords in the ReportCaster Repository. In Release 5.2.3, when ReportCaster is configured with Managed Reporting, you can integrate the Managed Reporting Servlet Exit with ReportCaster to authenticate user passwords in an external repository. Note that ReportCaster has utilities to bulk load existing Managed Reporting or external user IDs. For details about how to load user IDs into the ReportCaster Repository and configure the ReportCaster authentication exit, see *ReportCaster Authentication Exit* in the *WebFOCUS Custom Security Exits* chapter in the *WebFOCUS Security and Administration* manual.

ReportCaster User Credentials

Due to ReportCaster supporting multiple WebFOCUS Reporting Servers, there is a change in how ReportCaster utilizes user credentials. There is a ReportCaster owner ID (same as Managed Reporting user ID when ReportCaster is configured with Managed Reporting) for ownership of schedules, distribution lists, access lists, and Report Library content. In addition, an Execution ID specifies the user ID to run the scheduled job. This implementation permits users to access all schedules, distribution lists, access lists, and Report Library content independent of the server that executes scheduled jobs.

Configuring Access to the User Interfaces

By default, Managed Reporting Developers and Administrators are configured to access the ReportCaster Development and Administration Interface. Managed Reporting Analytical users can only access the ReportCaster Scheduling Wizard and HTML User Interface.

In Release 5.2.3, the Managed Reporting Applet Environment can be configured so that Managed Reporting Developers and Administrators access the ReportCaster Scheduling Wizard and HTML User Interface rather than the ReportCaster Development and Administration Interface.

You can change the interface by editing the following files located in /ibi_html/workbnch:

- mrdetectx.htm controls the Managed Reporting Developer user type.
- mrAdminFlags.js controls the Managed Reporting Administrator user type.

To configure the Developer or Administrator user type to access the ReportCaster Scheduling Wizard and HTML User Interface, set the following flag:

```
var WFInstallOption_ReportBrokerHTMLFlag=true
```

To configure the Developer or Administrator user type to access the ReportCaster Development and Administration Interface, set the flag back to the default setting:

```
var WFInstallOption_ReportBrokerHTMLFlag=false
```

If you are using the WebFOCUS Servlet, you must reload the WebFOCUS Web application in order for your changes to take effect. In addition, users must log out and log back in.

Development and Administration Interface

The ReportCaster Development and Administration Interface, which is a full client application that uses Java™ Web Start technology, replaces the Java applet user interface. From this redesigned interface, which requires JRE 1.4.0_02 (supplied with the installation CD) on the user's desktop, you can:

- Access the ReportCaster Server Configuration tool.
- Create users and groups to provide ReportCaster capabilities for non-Managed Reporting users.
- Create and maintain a schedule.
- Access the Library Access List and Library Management interfaces of the optional Report Library product.
- View a log report and maintain the log file.
- Create and maintain a Distribution List.

ReportCaster Server Configuration Tool

ReportCaster provides a GUI configuration tool for ease of configuration of ReportCaster. Using the ReportCaster Server Configuration tool, administrators can navigate through and change the various configuration parameter settings. For example, ReportCaster enables administrators to change the polling interval for the ReportCaster Distribution Server, define access to multiple WebFOCUS Reporting Servers, and configure the ReportCaster Distribution Server for National Language Support (NLS). For more information about how to access and use the ReportCaster Server Configuration tool, see the *ReportCaster Server Configuration* chapter in the *ReportCaster Development and Administration* manual.

Creating and Managing ReportCaster User IDs

When Managed Reporting is configured with ReportCaster, you must use the Managed Reporting User Administrator tool to create and manage users and groups. Managed Reporting calls ReportCaster and synchronizes user information in the ReportCaster Repository. When ReportCaster is not configured with Managed Reporting, you must use the User Administrator tool in the ReportCaster Development and Administration Interface to create and manage users and groups. The user ID created with the Managed Reporting or ReportCaster User Administrator tool is the user ID that owns objects (schedules, Distribution Lists, log reports, Library Access Lists, and Library Content). Note that you can now assign ReportCaster Administrator privileges to multiple ReportCaster user IDs.

For detailed information about creating and managing ReportCaster user IDs, see the *Accessing ReportCaster and Enabling User Capabilities* chapter in the *ReportCaster Development and Administration* manual.

Distribution Enhancements

In this section:

Distributing a Task

Distribution Enhancements When Creating a Schedule

Using the ReportCaster Development and Administration Interface, you can create and distribute multiple jobs (Tasks) within a single schedule that uses the e-mail or printer distribution methods. You can distribute one Task per schedule as an inline e-mail message. Additional Tasks are sent as e-mail attachments.

Distributing a Task

When creating a schedule from the ReportCaster Development and Administration Interface, you must create a Task to be distributed by ReportCaster. You can create the following types of Tasks:

- **WF Server Procedure.** Schedules the distribution of a WebFOCUS report that resides on a specified WebFOCUS Reporting Server.
- **Standard Report.** Schedules the distribution of a WebFOCUS report that resides in a Managed Reporting Standard Reports folder.
- **My Report.** Schedules the distribution of a WebFOCUS report that resides in a Managed Reporting My Reports folder.
- **URL.** Schedules the contents of a URL to specified recipients. The contents of a URL may be distributed through e-mail, using FTP, to a printer, or to the Report Library.
- **File.** Schedules the distribution of a file to which the ReportCaster Distribution Server has read access. When scheduling a file, you must enter the fully-qualified path and file name (for example, *d:\reportcaster52\filename.doc*) of the file. For example, if you want to distribute a Word document, you can send the static file to ReportCaster recipients.

The options for creating a Task differ depending on the Task type. For more information about these options, see the specific procedure for each Task type in *Creating a Task* in the *Creating and Maintaining a Schedule* chapter in the *ReportCaster Development and Administration* manual.

Distribution Enhancements When Creating a Schedule

When creating a schedule from the ReportCaster Development and Administration Interface, you can:

- Select from a list of Execution IDs you have previously used for the selected server. This new feature is available when scheduling jobs in Release 5.2.3.
- Schedule jobs to run within 1 to 59 minute intervals. The minute interval is useful when scheduling an alert.
- Support multiple WebFOCUS Reporting Servers to create and run schedules.
- Distribute output to the Report Library (see *Report Library* on page 5-9).
- Create distribution information dynamically when a scheduled job runs. This is accomplished by running a WebFOCUS procedure.
- Run a schedule once and delete it if it is not scheduled to run again.
- Send a single e-mail including multiple burst sections as attachments.
- Zip scheduled output (e-mail only).
- Specify a maximum of 256 characters of text within the body of an e-mail message.

For detailed information about the above enhancements, see the *Creating and Maintaining a Schedule* chapter in the *ReportCaster Development and Administration* manual.

- Distribute a report that is created using the WebFOCUS GRAPH FILE command. For more information about how to accomplish this, see *Using the GRAPH FILE Command* in the *Tips and Techniques for Coding a ReportCaster Report* appendix in the *ReportCaster Development and Administration* manual.
- Distribute a report that is created using the -HTMLFORM command. For more information about how to accomplish this, see *Using -HTMLFORM* in the *Tips and Techniques for Coding a ReportCaster Report* appendix in the *ReportCaster Development and Administration* manual.
- Specify additional WebFOCUS output formats (ALPHA, GIF, XML, TABT, COMMA, EXL97, and EXL2K FORMULA). For detailed information about these formats, see the *ReportCaster Formats* appendix in the *ReportCaster Development and Administration* manual.

Scheduling Wizard and HTML User Interface

In Release 5.2.3, Managed Reporting Analytical Users can create a schedule for Standard Reports or their own My Reports using the ReportCaster Scheduling Wizard. The ReportCaster Scheduling Wizard may be accessed from both the Business Intelligence Dashboard and Managed Reporting Domains environments. Once you have created the schedule, you can access the ReportCaster HTML User Interface to edit the properties of the schedule, delete the schedule, or run a log report to obtain information about the schedule. Additionally, you can purge log records to conserve space in the log file. These redesigned interfaces, which use DHTML technology, replace the ReportCaster Java applet user interface.

Distribution Enhancements When Using the Scheduling Wizard

When creating a schedule using the Scheduling Wizard, you can:

- Select from a list of Execution IDs you have previously used for the selected server. This new feature is available when scheduling jobs in Release 5.2.3.
- Distribute reports to the Report Library (see *Report Library* on page 5-9).
- Run a schedule once and delete it if it is not scheduled to run again.
- Specify additional WebFOCUS output formats (ALPHA, GIF, XML, TABT, COMMA, EXL97, and EXL2K FORMULA).
- Specify a maximum of 256 characters of text within the body of an e-mail message.

For detailed information about how to access and use the Scheduling Wizard and ReportCaster HTML User Interface, see the *ReportCaster End User's Manual*.

Report Library

When you create a schedule, you can specify to distribute schedule output to the Report Library, an optional storage and retrieval facility. The schedule output must be stored in an SQL repository (for example, SQL Server, Oracle, or DB2). The Report Library can contain any information that is distributed by ReportCaster (WF Server Procedures, Standard Reports, My Reports, URLs, and Files). When distributing to the Report Library, you can send an e-mail informing users of its availability and the link to the content in the library.

The Report Library includes secure access to library content, the ability to save multiple versions of the same output, and the ability to set an expiration date or keep a specified number of versions. The Report Library is only available to ReportCaster users who have been granted access to the library.

The Report Library consists of the following interfaces:

- **Library Access List.** Defines who may access specific content in the Report Library.
- **Library Content.** View the content in the Report Library to which you have been granted access.
- **Library Management.** ReportCaster Administrators can view a high-level summary of the content in the Report Library. Reports can be deleted, but the actual content of the reports cannot be viewed.

For detailed information about how to access and use the Report Library, see the *Report Library* chapter in the *ReportCaster Development and Administration* manual, or see the *Report Library* chapter in the *ReportCaster End User's Manual*.

ReportCaster Console

The ReportCaster Console is an interactive administrator's tool that may be used to maintain and view schedule and log information stored in the ReportCaster Repository. From the redesigned ReportCaster Console, you can select the following options:

- **Info.** Generates a list of schedules based on criteria you specify. The resulting list may then be used to view additional information about a specific schedule. You may also run a schedule adding Task parameters, and run a log report for a specific schedule.
- **Status.** Generates a list of scheduled jobs that are in the ReportCaster Distribution Server queue. Depending on the status of the job, you can then delete a job, change its priority, or run a Job Process Log Report.
- **Log.** Generates a list of schedules based on criteria you specify. The resulting list may then be used to view a Job Process Log Report, or purge a log transaction. These actions may be performed for a specific schedule, or for all schedules.
- **Execution ID.** Adds or deletes an Execution ID, which is a valid user ID that is used to run a scheduled Task on a specified server. You can also change the password of an Execution ID. In Release 5.2.3, you can change the Execution ID for specified owners on a specific server. For more information about how to accomplish this, see *How to Change an Execution ID* in the *ReportCaster Console* chapter in the *ReportCaster Development and Administration* manual.
- **Tools.** Globally replaces field values in the ReportCaster Repository.
- **Logoff.** Logs off the ReportCaster Console.
- **Help.** Opens the online help file.

For detailed information about how to access and use the ReportCaster Console, see the *ReportCaster Console* chapter in the *ReportCaster Development and Administration* manual.

Log Enhancements

ReportCaster enables you to:

- Suppress log file messages for burst values not distributed.
- Add burst values to log information indicating successful and unsuccessful distribution.
- Include distribution information by Task.

For more information about viewing a log report and purging log records, see *Tracking a Schedule Using the Schedule Log Option* in the *Creating and Maintaining a Schedule* chapter in the *ReportCaster Development and Administration* manual, or see *Viewing a Log Report* and *Purging the Log File* in the *Maintaining a Schedule* chapter in the *ReportCaster End User's Manual*.

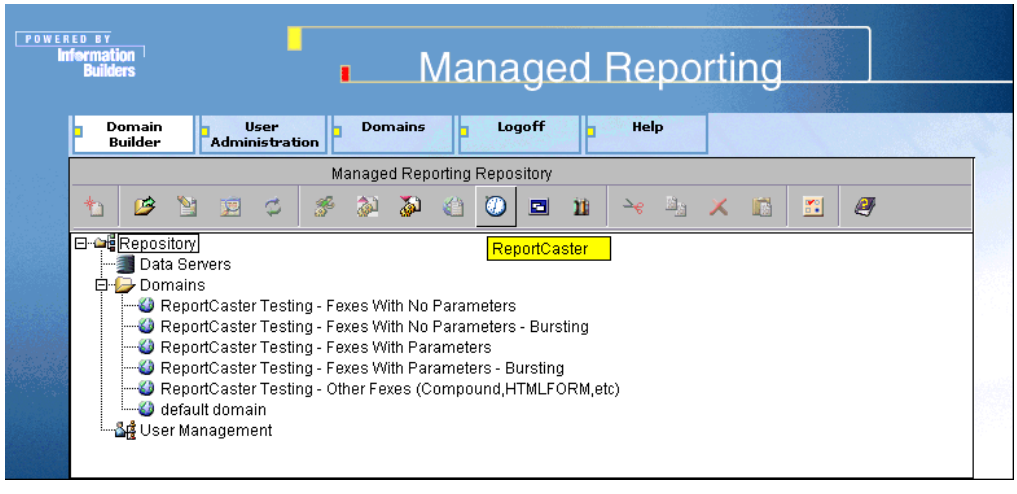
Configuring ReportCaster for NLS

You can use the ReportCaster Development and Administration Interface to configure the ReportCaster Distribution Server for National Language Support (NLS).

Initial configuration of NLS for WebFOCUS, including the ReportCaster Distribution Server, occurs when you install WebFOCUS and ReportCaster. The ReportCaster Development and Administration Interface is useful if you want to change code page values without reinstalling the product.

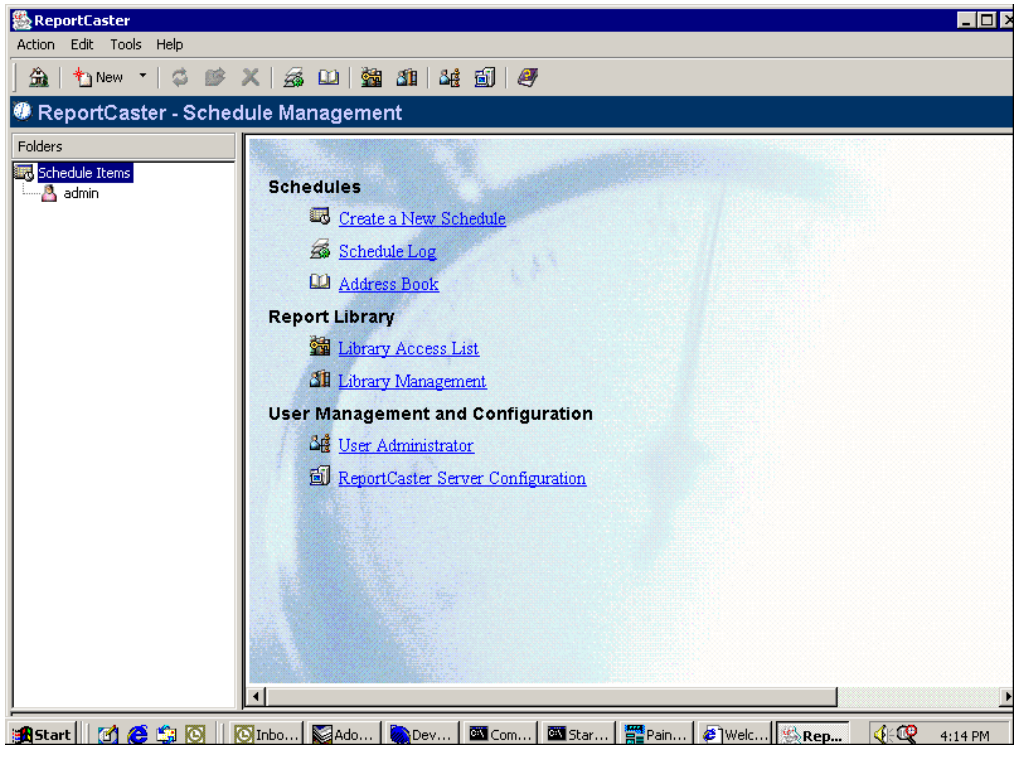
Procedure: How to How to Configure the ReportCaster Distribution Server for NLS

1. Launch WebFOCUS Managed Reporting.

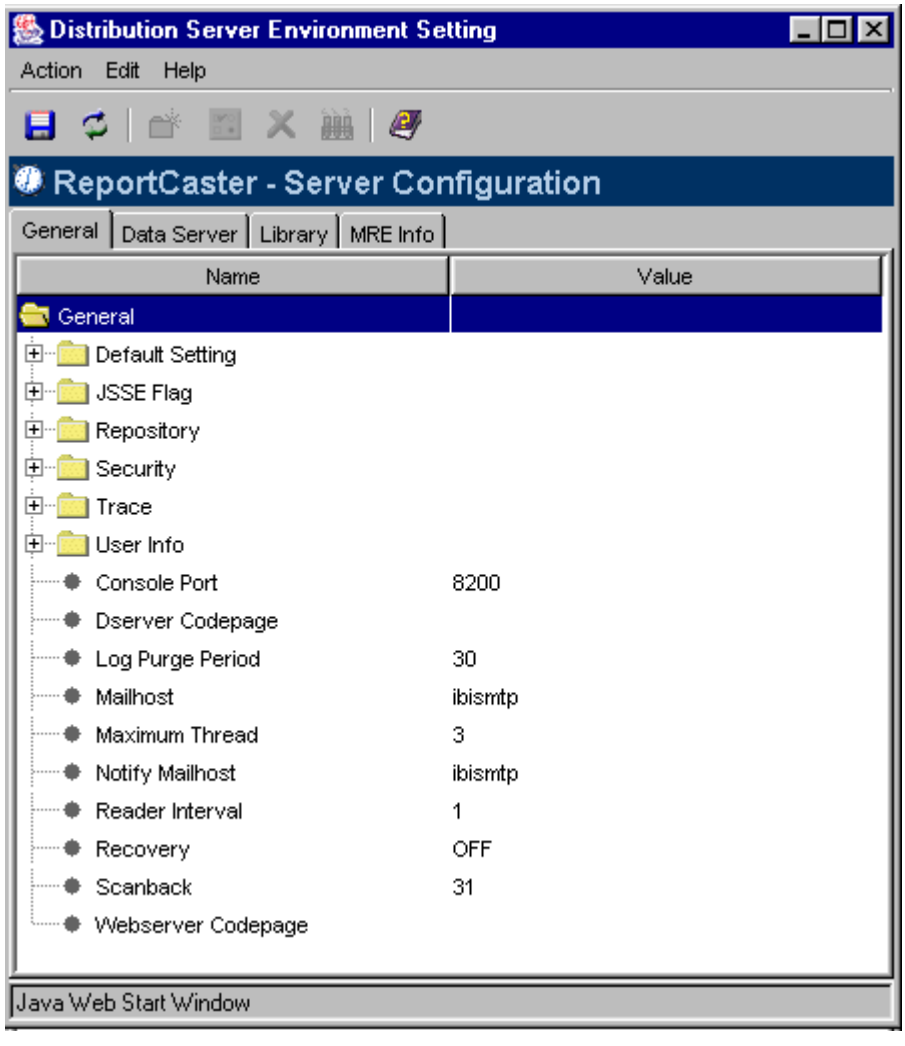


2. Click the *ReportCaster* icon.

The ReportCaster Development and Administration Interface opens:

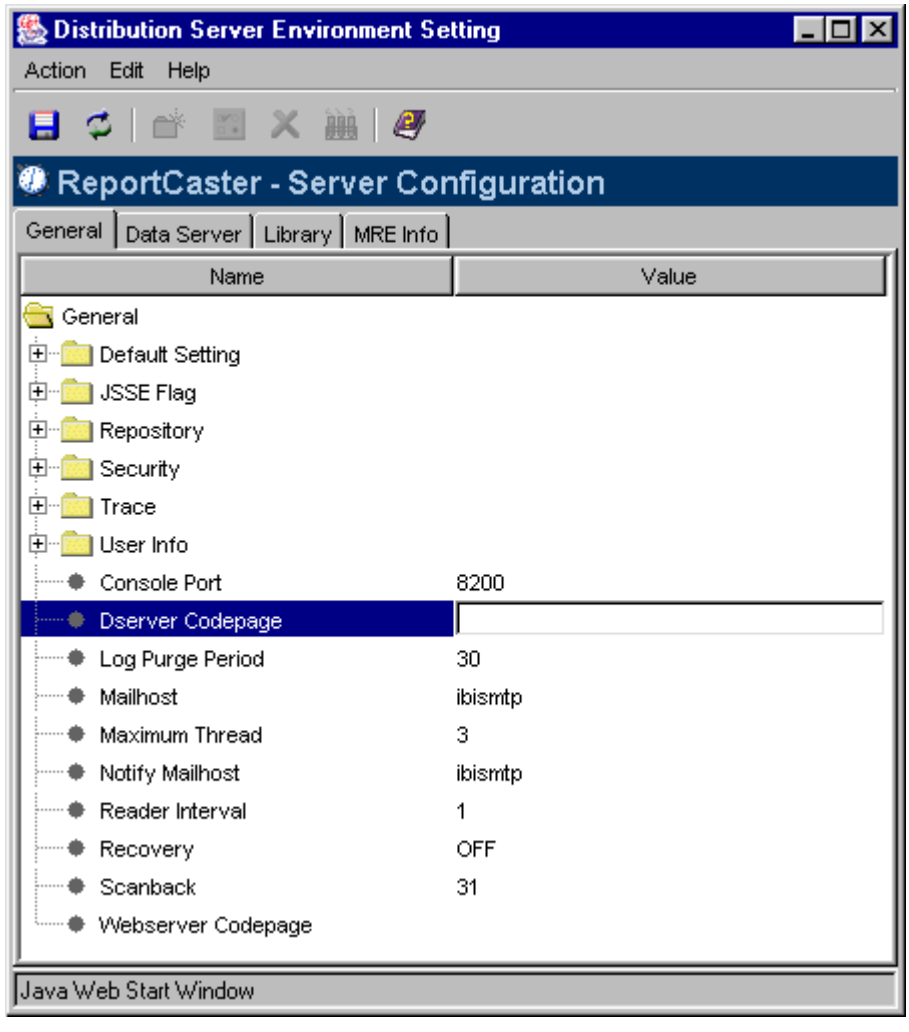


3. Click *ReportCaster Server Configuration*. The ReportCaster - Server Configuration window opens:



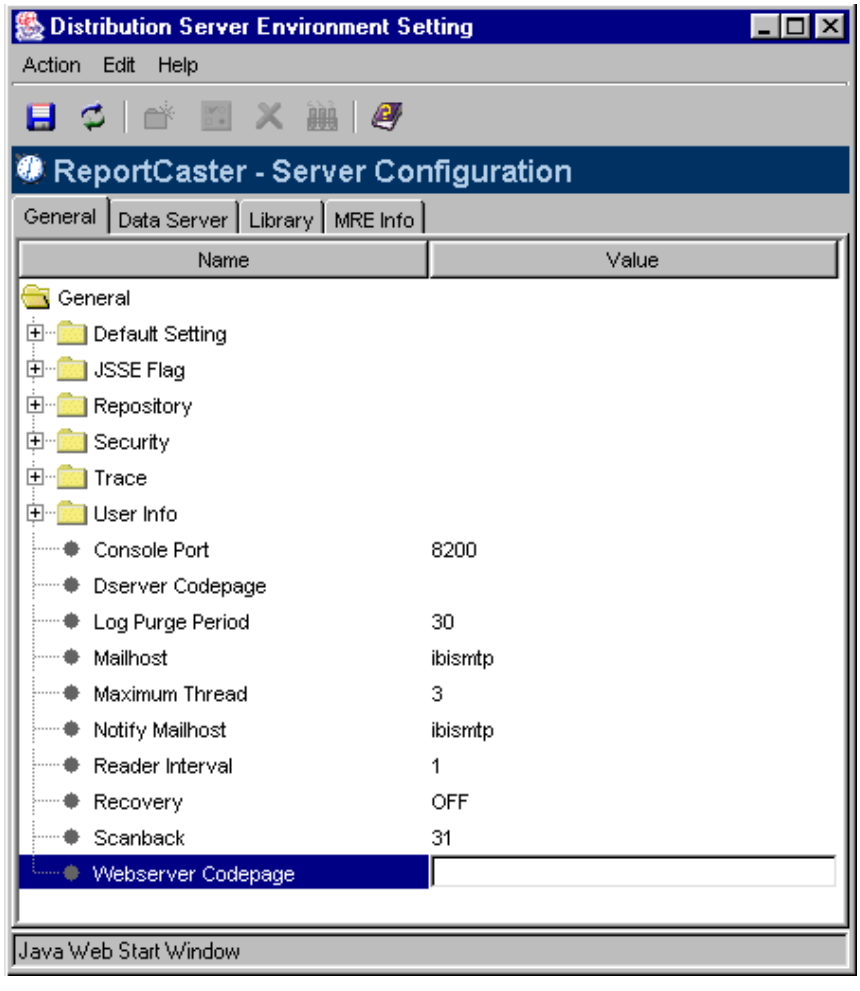
4. Select *Dserver Codepage* and enter a value.

   This value is based on the operating system on which the ReportCaster Distribution Server is running and the language you would like ReportCaster to use for report scheduling and distribution. For more information on code page values, contact your Information Builders representative.

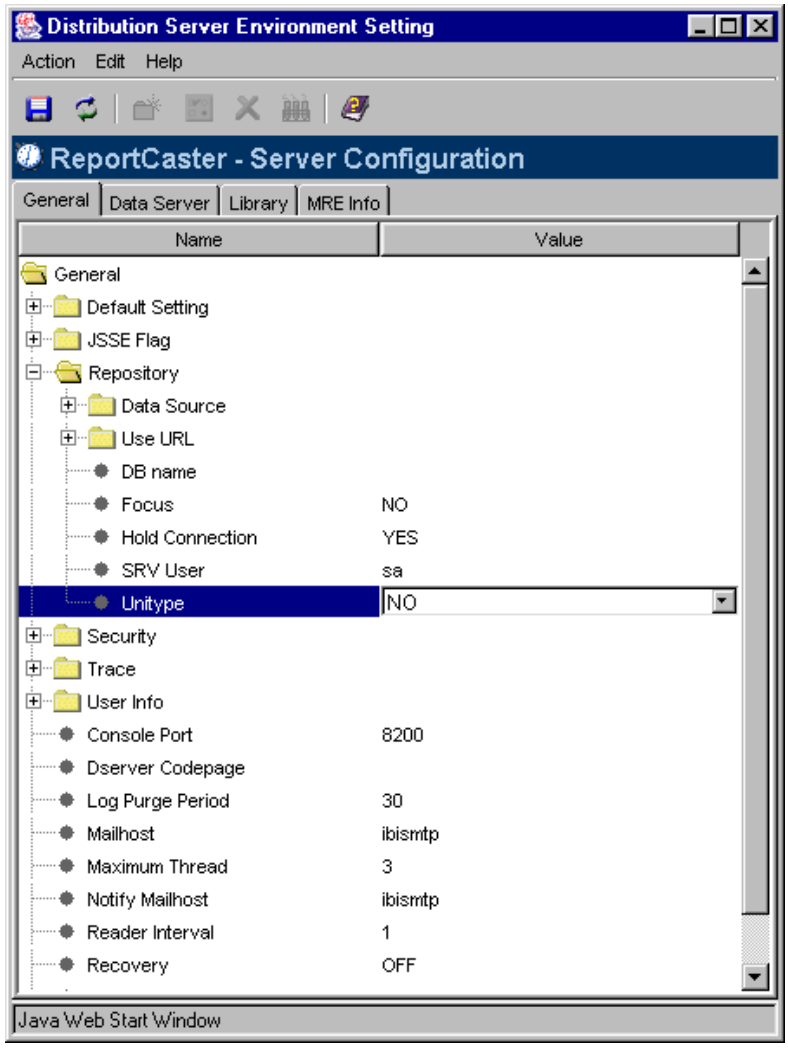


5. Select *Webserver Codepage* and enter a value.

   This value is based on the operating system on which the Web server is running and must be the language equivalent of the ReportCaster Distribution Server's code page. For more information on code page values, contact your Information Builders representative.



6. Open the *Repository* folder and select *Unitype*.



7. From the drop-down list, select either *YES* or *NO*.

   This setting activates Unitype for the Repository.

   If you are using Oracle or SQL Server repositories and require Unicode support, you must specify *Unitype = YES*. This setting is also required if your tables include Double-Byte Character Set (DBCS) characters.

8. Once you have configured the ReportCaster Distribution Server for NLS, click the *Save* icon.

ReportCaster API

The following are additional parameters for all components of the ReportCaster API:

- **ReportCaster User:** Owner of schedules and Distribution Lists.
- **Execution ID and password:** Valid credentials to connect to the server that runs the schedule.

The following is an additional parameter for the ReportCaster Bean API:

- **Server Name:** Must be specified since ReportCaster operates in a multi-server environment. Multiple server support using the ReportCaster Bean API will be introduced in a future release.

In Release 5.2.3, the following are additional features for the ReportCaster API:

- You must exist as a ReportCaster user ID prior to utilizing any API functionality.
- ReportCaster migration of API schedules from Release 4.3.6 to Release 5.2.3 will assign the WebFOCUS Reporting Server user ID and password in Release 4.3.6 as the owner ID/password and Execution ID/password. For an overview about ownership for the ReportCaster API components (Bean, Servlet, and Subroutine), see *ReportCaster API Authentication and Ownership* in the *Introducing the ReportCaster API* chapter in the *ReportCaster and Two-Way Email API for Self-Service Applications* manual.
- The ReportCaster Servlet API provides the option to assign ownership with the Default User (Release 5.2.1/5.2.2 compatible) or the WebFOCUS cookie/IBIB_user (Release 4.3.6 compatible). For more information about assigning ownership for the ReportCaster Servlet API, see *Servlet Security* in the *ReportCaster Servlet API* chapter in the *ReportCaster and Two-Way Email API for Self-Service Applications* manual.

Two-Way Email API

The following additional features concern Two-Way Email API authentication and customization, as well as accessing and editing sample Two-Way Email API JavaServer Pages™ (JSP™).

- **Authentication.** The Two-Way Email API authenticates users the same way as the ReportCaster Bean API. For more information, see the *ReportCaster and Two-Way Email API for Self-Service Applications* manual.
- **Customization.** You can customize the content of the Two-Way Email API Event Log by editing the Event Log setting in the ReportCaster Distribution Server configuration file. This setting controls what Two-Way Email inbox events should be logged. Possible values are:
  - ON displays all events.
  - ERROR displays only errors.
  - OFF turns off the Event Log. Nothing displays.

  You may edit the ReportCaster Distribution Server configuration file using the ReportCaster Server Configuration tool. For detailed information about this configuration tool, see the *ReportCaster Server Configuration* chapter in the *ReportCaster Development and Administration* manual.
- **Two-Way Email API JavaServer Pages:**

  To access the sample Two-Way Email API JavaServer Pages, enter the following URL:

  `http://hostname/rcaster/samples/twindex.htm`

  where:

  hostname

  Is the name of the ReportCaster Distribution Server.

  To edit the sample Two-Way Email API JavaServer Pages, go to the directory where the Web application is deployed or referenced by your servlet engine or application server.

CHAPTER 6

WebFOCUS Client Enhancements

This chapter describes all of the features for the WebFOCUS Client in Version 5 Release 2 and has been updated to reflect Version 5 Release 2.3. Topics that contain Version 5 Release 2.3 features are marked with asterisks.

Topics:

WebFOCUS Client Console

Introducing the WebFOCUS Client Console

* Configuring WebFOCUS Reporting Server Settings

* Configuring WebFOCUS Client Settings

WebFOCUS Client Tracing

Quick Links

National Language Support (NLS)

* National Language Support for International Computing

* Using the Dynamic Language Switch

Customizing the User Interface

Introducing the WebFOCUS Client Console

The WebFOCUS Client Console enables you to remotely manage your WebFOCUS environment. Using the WebFOCUS Client Console, administrators can navigate through and change the various configuration settings for the WebFOCUS Client. For example, administrators can modify default global settings, which are now located in cgivars.wfs instead of ibidir.wfs. Additionally, administrators can use the console to create WebFOCUS Client node profiles for each WebFOCUS Reporting Server.

Note: The following topics describe how to access and use the WebFOCUS Client Console. These topics include WebFOCUS Client settings that are available in Release 5.2. Information about all WebFOCUS Client settings that may be modified using the WebFOCUS Client Console is documented in the *WebFOCUS Administration Using the WebFOCUS Client Console* chapter in the *WebFOCUS Security and Administration* manual.

Accessing the WebFOCUS Client Console

You can access the WebFOCUS Client Console by entering the following URL:

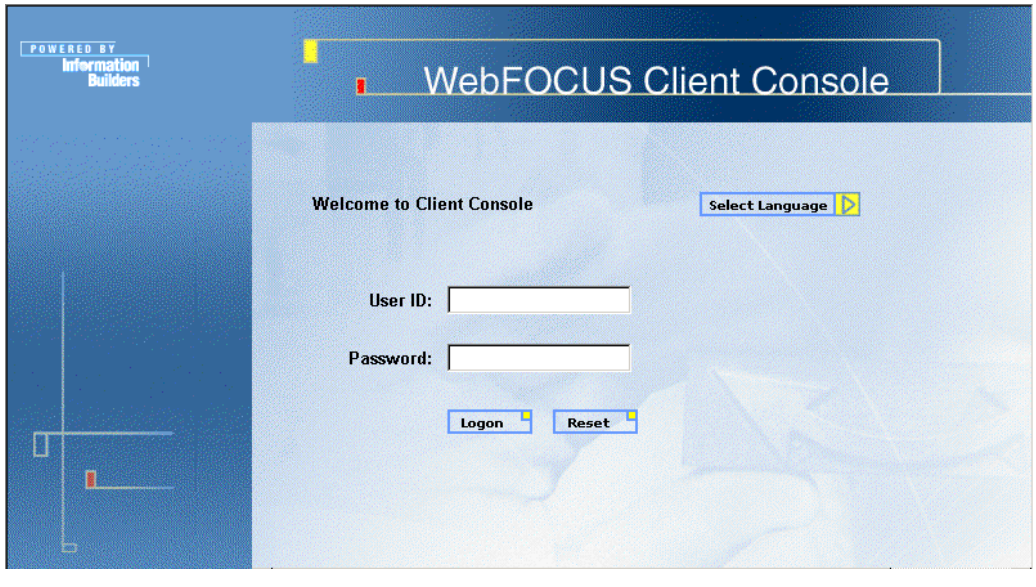
`http://hostname/ibi_html/wfconsole.htm`

where:

hostname

Is the host name of the machine on which the WebFOCUS Client is installed.

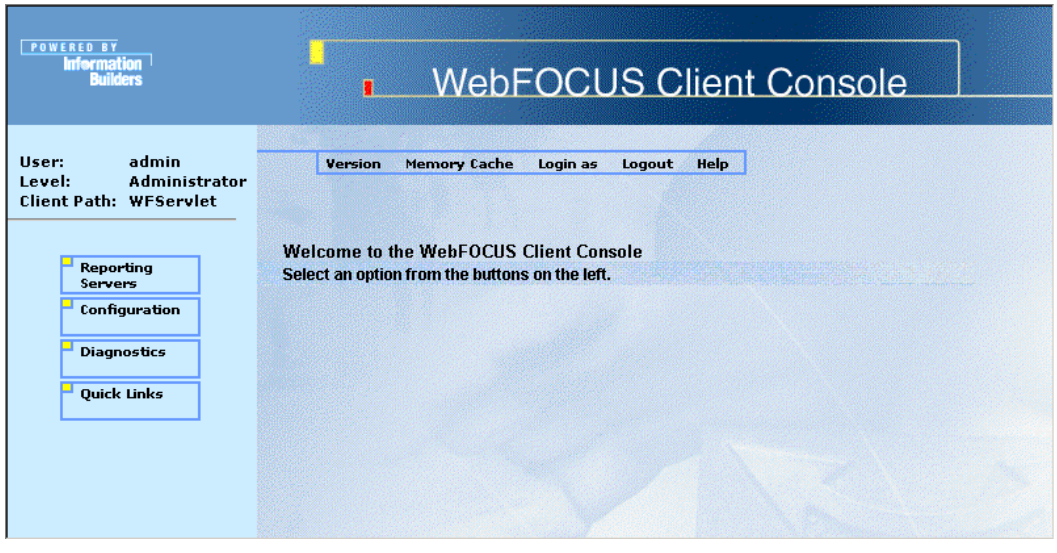
The WebFOCUS Client Console logon window appears:



Specify a valid user ID and password. If you are logging on to the WebFOCUS Client Console for the first time, specify the admin user ID. Once you have successfully logged on, you can change the type of authentication and IDs to use for all subsequent logons by specifying values in the IBIWFC_Authentication, ADMINISTRATORS, and DEVELOPERS settings in the ibiweb.cfg file. Developers may only access the tracing and Quick Links options in the WebFOCUS Client Console.

To view the WebFOCUS Client Console using a different language, click *Select Language* to display the available languages specified during the WebFOCUS installation. If only one language was selected during the installation, the Select Language option does not display.

After a successful logon, the welcome window appears:



Configuring WebFOCUS Reporting Server Settings

You can use the Reporting Servers section of the WebFOCUS Client Console to:

- Add or change server settings in the ibi/client52/wfc/etc/odin.cfg file.
- Configure Deferred Server Mappings in the ibi/client52/wfc/etc/altdnode.wfs file.
- Configure special settings for Server-Side Graphics and amper auto prompting in the ibi/client52/wfc/etc/odin.cfg file. This is a new feature in Release 5.2.3.

Adding or Changing Server Settings

How to:

Add a New Node to ODIN.CFG

Change a Node

Example:

Overriding WebFOCUS Default Settings for a Specific Server Node

Administrators can use the WebFOCUS Client Console to add WebFOCUS Reporting Servers and HTTP Listeners to the WebFOCUS environment. Additionally, you can use the console to add WebFOCUS Client node profiles for each WebFOCUS Reporting Server. The nodes are added to the ibi/client52/wfc/etc/odin.cfg file.

Procedure: How to Add a New Node to ODIN.CFG

1. Click *Reporting Servers* and then *Remote Services*.
2. Click *New*.
3. In the NODE field, specify the node name.
4. Specify the Node class (Client or Cluster).
5. Click *Next*.
   - If you selected Client in step 4, proceed to step 6.
   - If you selected Cluster in step 4, proceed to step 7.

6. The New Node block window appears. Enter the following required parameters:

   a. **PROTOCOL.** Rules for communicating between nodes (for example, TCP).

   b. **PORT.** Service (port number) or name.

   c. **HOST.** Name or IP address of the server.

   d. **CLASS.** Purpose of the node. For example, CLIENT or CLUSTER. If you specify CLIENT for a z/OS or OS/390 (MVS) server, you must include a qualifier.

   e. **SECURITY.** In Release 5.2.3, this is a new setting. Select one of the following options from the drop-down list:

      Explicit - A user ID and password are required to connect to this server node. Explicit is the default value.

      IWA (Integrated Windows Authentication) - This option is only available on Windows when using the WebFOCUS CGI. When IWA is set, no user ID or password is required to connect to this server node. Instead, the user context of the WebFOCUS Client connection is derived by the server using a Microsoft security API.

      Password Passthru - The user ID and password are explicitly specified for each connection and passed to the server, at connection time, for authentication. This option requires that the server be started with security off.

   Note: You can also specify the following optional parameters:

   - **COMPRESSION.** Turns on data compression. Codes are: 0 (off) and 1 (on).
   - **ENCRYPTION.** Sets data encryption ability.
   - **CONNECT_LIMIT.** Number of seconds the client holds the pending connection. This is useful in a cluster deployment to avoid a lengthy delay of failover response. Other possible values are 0 (no wait) and -1 (infinite wait). -1 is the default value.
   - **MAXWAIT.** <query wait>[,<row wait>]. Time the client waits before timeout. The first number is the return time for any row. The second number (optional) is the return time for rows beyond the first row. Time is in seconds.
   - **TYPE.** In Release 5.2.3, this is a new setting. Select one of the following options from the drop-down list:

     INTERNAL - Node is ignored by the Communication Gateway Server.

     RESERVED - Node is stopped and started by Cluster Manager to optimize performance.
   - **DESCRIPTION.** Description for the WebFOCUS Reporting Server node. This description displays in the WebFOCUS front-end tools.

   Proceed to step 8.

7. The New Cluster Node block window appears. Enter the following required parameters:
   a. **ALTERNATE.** Select the servers to be included in the cluster.
   b. **DESCRIPTION.** Description for the cluster.
   c. **LICENSE.** License key. If a license key has not already been specified, add a license key that will enable the cluster feature.
8. Click *Save*.

Procedure: How to Change a Node

1. Click *Reporting Servers* and then *Remote Services*.
2. Select the node you want to change.
3. Click one of the following buttons:
   - **Modify.** Displays the settings for the selected node, enabling you to make changes.
   - **Remove.** Deletes the selected node. You will receive a message asking for you to confirm the deletion.
   - **Profile.** Enables you to override WebFOCUS default settings for a specific WebFOCUS Reporting Server node. These settings are written to ibi/client52/wfc/etc/*node*.prf, where *node* is the node you selected in step 2. For examples, see *Overriding WebFOCUS Default Settings for a Specific Server Node* on page 6-7.
   - **Server Console.** Displays the WebFOCUS Reporting Server Console, which enables you to remotely manage your server environment. For more information, see the *iWay Server Administration for UNIX, Windows, OpenVMS, OS/400, OS/390, and z/OS* manual.

   Note: In Release 5.2.3, you can select the *Sort alphabetically* check box to sort a list of multiple servers.

Example: Overriding WebFOCUS Default Settings for a Specific Server Node

To override WebFOCUS default settings for a specific WebFOCUS Reporting Server node, set the IBI_REPORT_USER and IBI_REPORT_PASS settings to values that are appropriate for your environment.

In this example, an already authenticated Web server user ID is used for the connection to the WebFOCUS Reporting Server, and no password is required. This example is not supported when connecting to a Windows-based WebFOCUS Reporting Server.

```
IBI_REPORT_USER=&REMOTE_USER
IBI_REPORT_PASS="TRUSTED"
```

In this example, a valid user ID and password are required to connect to the WebFOCUS Reporting Server.

```
IBI_REPORT_USER=wfuser
IBI_REPORT_PASS=wfpass
```

In this example, you are dynamically prompted for server credentials.

```
IBI_REPORT_USER=
IBI_REPORT_PASS=
```

In this example, a user ID (chuck) and password (hill) for connecting to the WebFOCUS Reporting Server has been set up. They will only be used if a public user connects to the server from Managed Reporting or Dashboard.

```
<if> IBIMR_user.upper EQ "PUBLIC"
IBI_REPORT_USER=chuck
IBI_REPORT_PASS=hill
<endif>
```

Configuring Deferred Server Mappings

How to:

Map a Deferred Server Node to an Immediate Server Node

You can configure alternate server nodes for use with Managed Reporting's Deferred Receipt feature. For more information about Deferred Receipt, see the *WebFOCUS Managed Reporting Administrator's Manual*.

Deferred Receipt requests can be processed by using the immediate WebFOCUS Reporting Server (Immediate Server) or by using an Alternate Deferred Receipt Server (Deferred Server) dedicated to running only deferred requests. The resources for the Deferred Server are managed independently from the immediate Server. The Deferred Server must have the same access to applications, data sources, and Master Files, and run in the same environment (for example, UNIX), as the immediate Server.

Note: z/OS and OS/390 require deferred requests to run on an Alternate Server.

Procedure: How to Map a Deferred Server Node to an Immediate Server Node

Using the WebFOCUS Client Console, add a node for the Deferred Server the same way as adding a non-deferred node (see *Add a New Node to ODIN.CFG* on page 6-4). Next, perform the following steps to map the Deferred Server to an immediate Server node:

1. Click *Reporting Servers* and then *Deferred Server Mappings*.

   A page appears for editing the Deferred Server Mappings file (altdnode.wfs).

2. Click *New* to create a new mapping.
3. Specify the immediate Server in the Select Server drop-down list, which displays all nodes specified in the ibi/client52/wfc/etc/odin.cfg file.
4. Specify the Deferred Server in the Select Deferred Server drop-down list, which displays all nodes specified in the ibi/client52/wfc/etc/odin.cfg file (excluding the immediate Server you just specified).
5. Click *Save*.

Note: You can map multiple immediate Servers to the same Deferred Server by repeating these steps.

Configuring Special Services

To use Server-Side Graphics or amper auto prompting when the WebFOCUS Client (CGI or ISAPI only) is installed on a Windows XP or Windows 2000 server platform, you must perform the following steps:

1. Configure the service and classpath for JSCOM3.
   a. Click *Reporting Servers* and then *Special Services*. The New JSCOM3 Node window appears:

   b. In the SERVICE field, enter the port number on which you want the JSCOM3 node to listen.
   c. In the CLASS_PATH field, enter the following without leaving any spaces:

```
drive:/ibi/WebFOCUS52/ibi_html/javaassist/WFServlet.jar:
drive:/ibi/WebFOCUS52/ibi_html/javaassist/WFAppPass.jar:
drive:/ibi/WebFOCUS52/ibi_html/javaassist/IBIGifGraphChart.jar
```

      where:

      `drive` is the letter of the drive on which WebFOCUS is installed.

   d. Click *Save*.
2. Enable the JSCOM3 setting in the cgivars.wfs file.
   a. Click *Configuration* and then *Client Settings*. The settings in the cgivars.wfs file appear.
   b. Set the IBI_USE_JSCOM3 setting to *YES*.
   c. Click *Save*.
3. Start the WebFOCUS Client WorkSpace located under your WebFOCUS 52 Client Programs menu. This is separate from the WebFOCUS Reporting Server WorkSpace.

   For example, select *WebFOCUS52 Client* from Programs, and then click *Start Security ON*.

Configuring WebFOCUS Client Settings

You can use the Configuration section of the WebFOCUS Client Console to:

- Configure WebFOCUS startup parameters in the \ibi\client52\wfc\web\cgi\ibiweb.cfg file.
- Change WebFOCUS Client settings in the \ibi\client52\wfc\etc\cgivars.wfs file.
- Redirect report output using the \ibi\client52\wfc\etc\mime.wfs file.
- Customize WebFOCUS Client settings in the \ibi\client52\wfc\etc\site.wfs file.
- Configure National Language Support (NLS) settings in the \ibi\client52\wfc\etc\nlscfg.err file.

Configuring WebFOCUS Startup Parameters

The initial configuration file loaded by the WebFOCUS Client is ibiweb.cfg. The WebFOCUS installation program only inserts edaconf, ADMINISTRATORS, IBIWFC_Authentication, and DEVELOPERS into ibiweb.cfg. All additional settings must be added using the WebFOCUS Client Console.

To add and change settings in the ibiweb.cfg file, click *Configuration* and then *Startup Parameters*.

The following settings may be added, modified, or deleted in the ibiweb.cfg file:

Setting	Usage	Notes
IBIWFC_Authentication=[NONE|EDA|EDA:*EDANODE*|WEB]	[NONE|EDA|EDA:*EDANODE*|WEB] Optionally specifies the authentication method used to verify the administrator or developer user IDs.	NONE is the default value, which means that no authentication is desired. To indicate authentication against a particular EDANODE, set the parameter value as EDA:*EDANODE*. Only one EDANODE may be listed. The WEB setting indicates Web server authentication. If using the WFServlet, J2EE role based security is required for the WEB setting. **Note:** In Release 5.2.1/5.2.2, if you change this value, you must also change the value for the *WFInstallOption_Authentication* setting in ibi/WebFOCUS52/ibi_html/javaassist/cgipath.js.

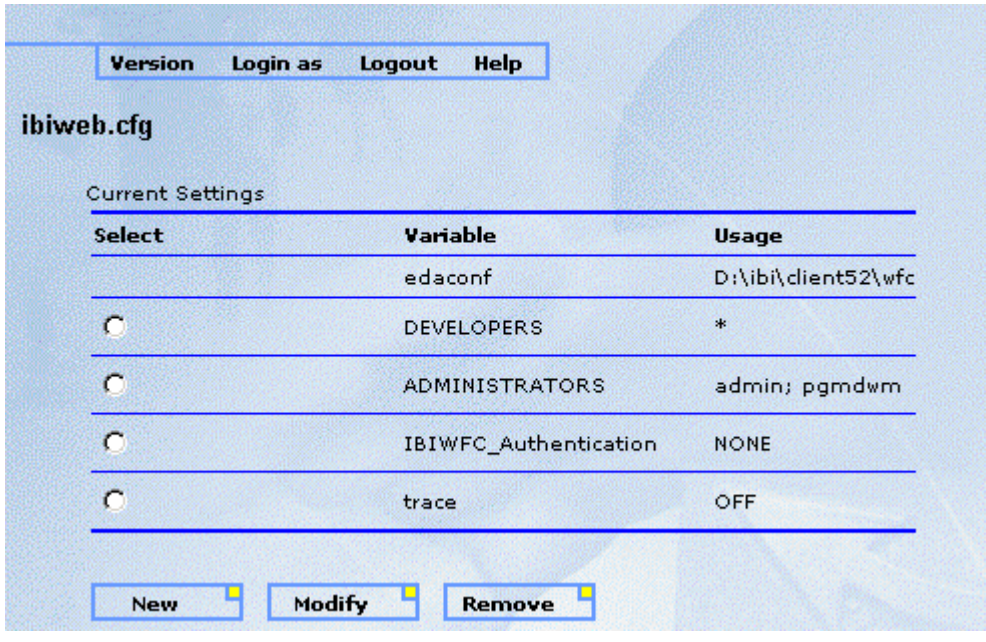
Setting	Usage	Notes
DEVELOPERS = [*|dev1;dev2]	[*|dev1;dev2] where: * Signifies that any user ID is valid. An asterisk (*) is the default value. dev1;dev2 Are the user IDs that can issue application commands. For more information about application commands, see the *Developing Reporting Applications* manual.	This setting controls certain Developer Studio features, such as the ability to use the Deploy Wizard. If you change this setting from an * to a list of developer user IDs, then these features are available only to the specified developer user IDs. Developers can set these WebFOCUS Client credentials under the WebFOCUS Environments properties in Developer Studio. You can set a default value during the Developer Studio installation. If set during the Developer Studio installation, this user ID must be a user ID that can connect to a WebFOCUS Reporting Server. This setting replaces the app_admin setting in ibidir.wfs. Authentication for a developer is applied in the same manner as for an administrator (for more information, see the ADMINISTRATORS setting).

Setting	Usage	Notes
`ADMINISTRATORS = admin1;admin2`	*admin1;admin2* Are the user IDs that can issue configuration commands and application commands. For more information about valid application commands, see the *Developing Reporting Applications* manual.	admin is the default value. If IBIWFC_Authentication is set to EDA and an EDANODE has not been specified, then any user IDs stored in the ADMINISTRATORS parameter must be able to connect to the default WebFOCUS Reporting Server (IBI_REPORT_SERVER). If an EDANODE has been specified, then any user IDs stored in the ADMINISTRATORS parameter must be able to connect to that EDANODE. If IBIWFC_Authentication is set to WEB, then any user IDs stored in the ADMINISTRATORS parameter must be able to connect to the Web server specified in the WebFOCUS configuration. If IBIWFC_Authentication is not set in ibiweb.cfg, then any user ID can be stored as an administrator.

Setting	Usage	Notes
MR_PGM_FLAGS = *positions1through10*	Position 5: Enables the MRCEXT3 exit, which allows for an external encryption algorithm to be used for WebFOCUS encryption. Position 7: Enables the MRCEXT4 exit, which allows for the external authentication of a user's credentials during a Managed Reporting signon. Position 8: Enables the MRCEXT5 exit, which allows for the external management of users. MRCEXT5 is used in conjunction with the MRCEXT1 exit. Position 9: Turns off the prompting of variables in Managed Reporting. Position 10: Enables WebFOCUS CGI callable exits to be invoked.	In Release 5.2.3, this setting may be modified using the WebFOCUS Client Console (CGI only). MR_PGM_FLAGS are positional flags that can be used to enable WebFOCUS custom security exits and to turn off the prompting of variables in Managed Reporting. When you select a Position check box, it places a Y in that position in MR_PGM_FLAGS. Positions 1-4 and 6 are reserved for internal use. You can select multiple Position check boxes. **Examples:** To use the MRCEXT4 exit and turn off variable prompting, select Position 7 and Position 9. The following appears in ibiweb.cfg: MR_PGM_FLAGS=------Y-Y- To use the MRCXT3 exit along with the MRCEXT4 exit, select Position 5 and Position 7. The following appears in ibiweb.cfg: MR_PGM_FLAGS=----Y-Y---

Example: Configuring IBIWEB.CFG

The following example depicts configuration settings that may appear in ibiweb.cfg:



Changing WebFOCUS Client Settings

All default global settings generated by installing WebFOCUS are located in \ibi\client52\wfc\etc\cgivars.wfs. To change the default global settings in the cgivars.wfs file, click *Configuration* and then *Client Settings*. The settings in the cgivars.wfs file appear. Make the necessary changes to the file and then click *Save*.

The following settings in cgivars.wfs may be changed. Those marked with an asterisk were added in Release 5.2.3.

Settings	Description
APPROOT	Location of the Application Namespace root directory on the Web server where WebFOCUS is installed. The default directory is *installationdrive*:\ibi\apps.
*FOCHTMLURL	Alias used to locate WebFOCUS resources on the Web server.
DISPLAY	Points to where the X-Windows server is located. This displays Server-Side Graphics on UNIX.

Settings	Description
IBIJAVACMD	Points to where the Java Virtual Machine (JVM) is located in order to execute Java language-based programs such as Server-Side Graphics.
WF_SIGNON_MESSAGE	Points to the HTML page that appears after a successful signon occurs.
*lang	This setting is applicable only when using the WebFOCUS CGI on UNIX. In order to display NLS characters on a report with Server-Side Graphics, you must set this parameter to the appropriate UNIX locale encoding (for example, en_us).
IBIF_external_xsl	Holds the URL to the XSL document used to transform the XML document produced by IBIF_internal_xsl into the desired final document (HTML form). If the variable is blank, then the resulting document from the internal transformation is sent to the client.
IBIF_webapp	The alias of the WebFOCUS Web application.
IBIF_wfdescribe	Variable for amper auto prompting. Possible values are: • XMLRUN - An XML document is created describing the amper variables in the procedure, and the procedure is executed. • XML - An XML document is created describing the amper variables. • OFF - Turns auto prompting off. OFF is the default value. This variable is always turned OFF for Managed Reporting requests.
IBIF_xsl	If the original XML document produced by the WebFOCUS Reporting Server was not type SYSFEX, the WebFOCUS Client will not perform internal and external transformations. The WebFOCUS Client will instead perform one transformation using the XSL document pointed to by the URL stored in this variable. This setting is blank by default.

Settings	Description
IBIWF_language	The language in which your WebFOCUS environment is running.
*WF_AUTOSIGNON	Auto prompting of server credentials. Possible values are: • YES - Displays a dynamic server signon page after a failed connection to the WebFOCUS Reporting Server. YES is the default value. • NO - Disables the dynamic server signon page. Users will receive a message after a failed connection to the WebFOCUS Reporting Server. • PREPROMPT - Displays a dynamic server signon page before attempting to connect to the WebFOCUS Reporting Server for the first time. This is a new setting in Release 5.2.3.
IBI_RES_ANALYZER	Specifies whether WebFOCUS Resource Analyzer is installed.
IBIF_persistentamp	Turns on the persistent && feature. YES is the default value. To disable, set to NO.
*ENCRYPT	If you select the ENCRYPT check box, the cgivars.wfs file will be encrypted. When you access the file again using the console, it will be decrypted for you to view, and the ENCRYPT check box will still be selected. If you edit the cgivars.wfs file outside of the console, its contents will be encrypted. In Release 5.2.3, you can encrypt the cgivars.wfs file using the WebFOCUS Client Console.

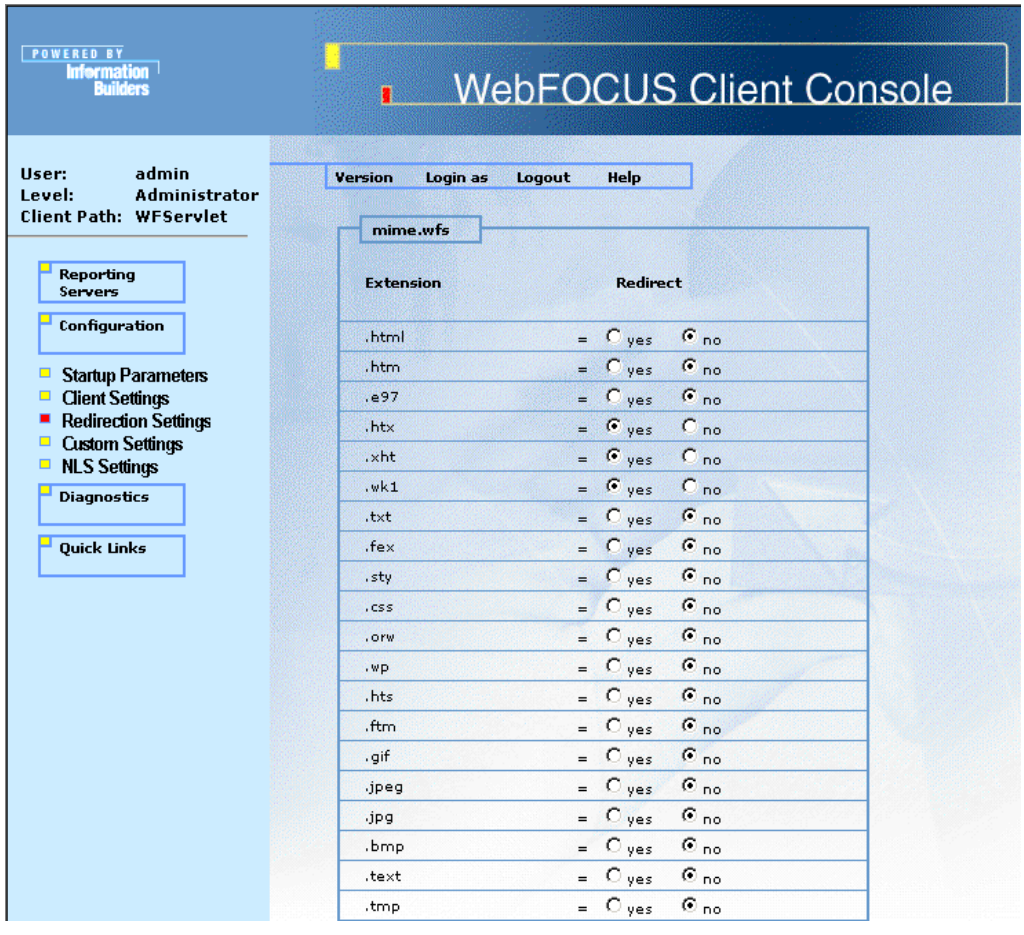
Redirection Settings

The mime.wfs file contains information about format types available with WebFOCUS. The following is a sample mime.wfs file:

```
<VER 1>

<!   file extension
<!   content type
<!   format of data (ascii/binary)
<!   use redirection (yes/no)
<!   convert to ascii (yes/no) used by EBCDIC base systems.
<!
<!         extension  content_type              format redirect translate
<!
<ADDTYPE> .html       text/html                 ascii   no        no
<ADDTYPE> .htm        text/html                 ascii   no        no
<ADDTYPE> .e97        application/vnd.ms-excel  ascii   no        no
<ADDTYPE> .htx        application/vnd.ms-excel  ascii   yes       no
<ADDTYPE> .xht        application/vnd.ms-excel  ascii   yes       &astran
<ADDTYPE> .wk1        application/vnd.ms-excel  binary  yes       no
<ADDTYPE> .txt        text/plain                ascii   no        no
<ADDTYPE> .fex        text/plain                ascii   no        no
<ADDTYPE> .sty        text/plain                ascii   no        no
<ADDTYPE> .css        text/css                  ascii   no        no
<ADDTYPE> .orw        application/x-odin-int    binary  no        no
<ADDTYPE> .wp         text/plain                ascii   no        no
<ADDTYPE> .hts        text/plain                ascii   no        no
<ADDTYPE> .ftm        application/x-ftm         ascii   no        no
<ADDTYPE> .gif        image/gif                 binary  no        no
<ADDTYPE> .jpeg       image/jpeg                binary  no        no
<ADDTYPE> .jpg        image/jpeg                binary  no        no
<ADDTYPE> .bmp        image/x-MS-bmp            binary  no        no
<ADDTYPE> .text       text/plain                ascii   no        no
<ADDTYPE> .tmp        text/plain                binary  no        no
<ADDTYPE> .mas        text/plain                ascii   no        no
<ADDTYPE> .xml        text/xml                  ascii   no        no
<ADDTYPE> .xls        application/vnd.ms-excel  binary  yes       no
<ADDTYPE> .ps         application/postscript    ascii   no        &astran
<ADDTYPE> .prn        application/x-prn         ascii   no        &astran
<ADDTYPE> .pdf        application/pdf           ascii   yes       &astran
<ADDTYPE> .dif        application/x-dif         ascii   yes       &astran
<ADDTYPE> .gfa        application/gfa           binary  no        no

<ADDTYPE> .doc        application/msword        ascii   yes       &astran

<IF> IBIMR_returntype contains "sync"
<ADDTYPE> .doc        text/plain                ascii   no        no
<ENDIF>
```

Setting the redirect column to yes enables redirection, which allows you to save the report output in a temporary directory. Then, an http call is made from the browser to retrieve the temporary stored output for display in the browser. If redirection is turned off, the report output displays in the browser immediately after the request is executed.

To access the redirect setting using the WebFOCUS Client Console, click *Configuration* and then *Redirection Settings*.



Change the Redirect settings where applicable, and then click *Save* to save your changes.

Note: If you select the *ENCRYPT* check box, the mime.wfs file will be encrypted. When you access the file again using the console, it will be decrypted for you to view, and the ENCRYPT check box will still be selected. If you edit the mime.wfs file outside of the console, its contents will be encrypted. In Release 5.2.3, you can encrypt the mime.wfs file using the WebFOCUS Client Console.

Customizing WebFOCUS Client Settings

You can customize the default global settings for WebFOCUS using the \ibi\client52\wfc\etc\site.wfs file. You would use site.wfs to override cgivars.wfs for a specific user or request.

To override the default global settings in the cgivars.wfs file, click *Configuration* and then *Custom Settings*. The settings in the site.wfs file appear. Make the necessary changes (after the <VER 1> tag) and then click *Save*.

The following additional setting may be added to site.wfs:

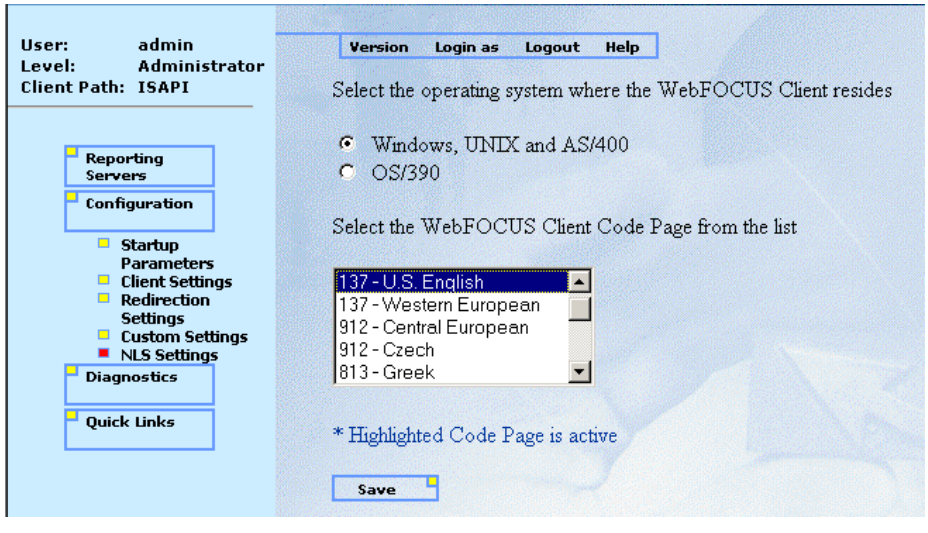
- **SIGNON_CORRECT.** The HTML page that appears after a successful signon.

In addition, if you select the *ENCRYPT* check box, the site.wfs file will be encrypted. When you access the file again using the console, it will be decrypted for you to view, and the ENCRYPT check box will still be selected. If you edit the site.wfs file outside of the console, its contents will be encrypted. In Release 5.2.3, you can encrypt the site.wfs file using the WebFOCUS Client Console.

NLS Settings

In Release 5.2.3, the interface for the NLS Settings option has been enhanced. To configure National Language Support (NLS) settings for the WebFOCUS Client:

1. Click *Configuration* and then *NLS Settings*.



2. Select the operating system on which the WebFOCUS Client is running.

   The available code pages for the selected operating system are displayed in the scroll-down list.

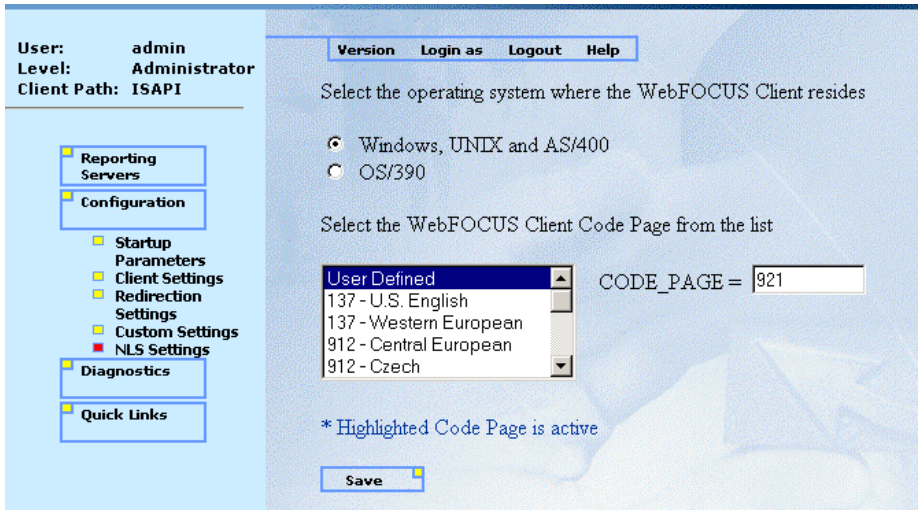
3. From the list, select a code page that configures the client for the correct display of report output in the browser.

   Tip: The language selected for the client usually corresponds to the language selected for the server from the WebFOCUS Reporting Server Console.

   If the language chosen from the WebFOCUS Reporting Server Console does not appear in the drop-down list on the WebFOCUS Client Console, select *User Defined* and manually enter the code page as prompted.

   You would do this, for example, if the server adds support for a new code page that is not yet reflected in the client software.

   In the following sample configuration window, the administrator entered code page 921 (Baltic Multilanguage):



4. Click *Save* to store your NLS settings. The console will generate and update the client configuration file (nlscfg.err) with the CODE_PAGE setting. Note that if you click *NLS Settings* again, the new setting is highlighted as the active code page.

WebFOCUS Client Tracing

The Diagnostics option in the WebFOCUS Client Console enables you to view and delete WebFOCUS Client trace files. Trace files enable you to track requests processed by the WebFOCUS Client (CGI, ISAPI, or Servlet). In order to view WebFOCUS Client trace files, you must set the trace parameter ON in the ibiweb.cfg file. You must also specify the location where you want the trace files to be sent using the trace_location parameter in ibiweb.cfg.

To view WebFOCUS Client trace files, click *Diagnostics* and then *Client Traces*. The current WebFOCUS Client trace files appear:

The Trace File column lists the WebFOCUS generated number identifying each trace. A WebFOCUS Servlet trace file is denoted by srv*nnnn*.trc, while a WebFOCUS CGI or ISAPI trace file is denoted by cg0000*nnnn*.trc. The Size column lists the size in bytes of each trace file, and the Query String column lists the actual request that was sent to the WebFOCUS Client. You can view or delete a trace file by selecting the file and then clicking the appropriate action button (View, Delete, or Delete All). When you click View, the trace file appears in a separate browser window, increasing your search speed without interrupting the console session.

Quick Links

The following Quick Links are available in the WebFOCUS Client Console:

- Information Builders - Provides a link to Information Builders' home page.
- WebFOCUS - Provides a link to the WebFOCUS home page, which includes utilities to run a procedure or ad hoc request, and the New Century Corp Self Service application (if the application was installed).

National Language Support for International Computing

National Language Support (NLS) ensures that the WebFOCUS Client correctly processes and displays all national characters embedded in any data source. It allows you to represent data in the formatting that corresponds to the currently selected user locale.

The WebFOCUS Client provides:

- Dynamic configuration for NLS on Windows NT/2000/XP, UNIX, OS/400, OS/390, and z/OS through the WebFOCUS Client Console. For more information, see *NLS Settings* on page 6-20.
- Inclusion of all supported localized versions in a single installation procedure. A dialog box prompts you to select one or more languages to install. For more information, see *Using the Dynamic Language Switch* on page 6-24.
- For localized versions, a toggle feature that enables an individual user to switch languages from any WebFOCUS logon page, such as the Managed Reporting logon page. The user can switch to any language selected during the installation procedure. Text in the user interface (including the logon page) then appears in the selected language. For more information, see *Using the Dynamic Language Switch* on page 6-24.
- Customization of the user interface through resource files. For more information, see *Customizing the User Interface* on page 6-30.
- In Release 5.2.3, support for Double-Byte Character Set (DBCS) code pages, in addition to the following translated user interfaces: Japanese, Korean, Simplified Chinese, and Traditional Chinese.

Using the Dynamic Language Switch

In this section:

Dynamic Language Switch Feature

Directory Structures

Editing Language Values

Reference:

Values for Supported Languages

All supported localized versions are included in a single WebFOCUS Client installation. In Release 5.2.3, all language files are installed by default so that you have the option of enabling a language at a later time. Only the languages you select during installation will be enabled initially.

You can dynamically switch between languages used throughout the user interface without affecting other users.

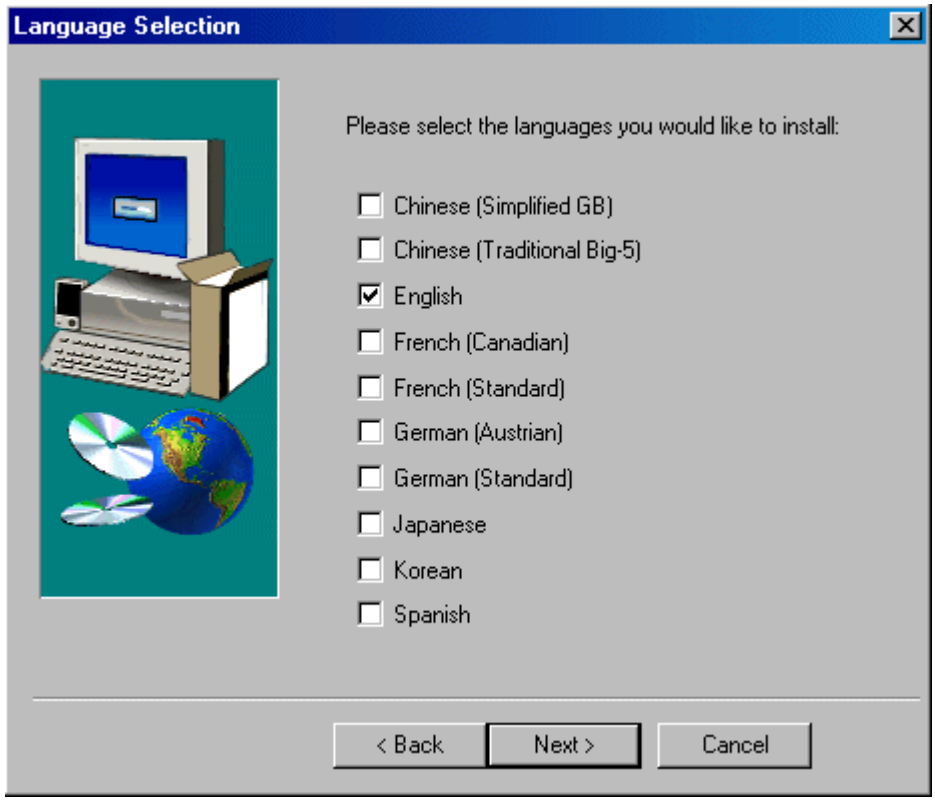
The installation program uses the same language specified in the operating system's regional settings or browser language options. If the language is not available in a localized version, English is used as the default.

In Release 5.2.3, you can install the following languages:

- Chinese (Simplified GB)
- Chinese (Traditional Big-5)
- English
- French (Canadian)
- French (Standard)
- German (Austrian)
- German (Standard)
- Japanese
- Korean
- Spanish

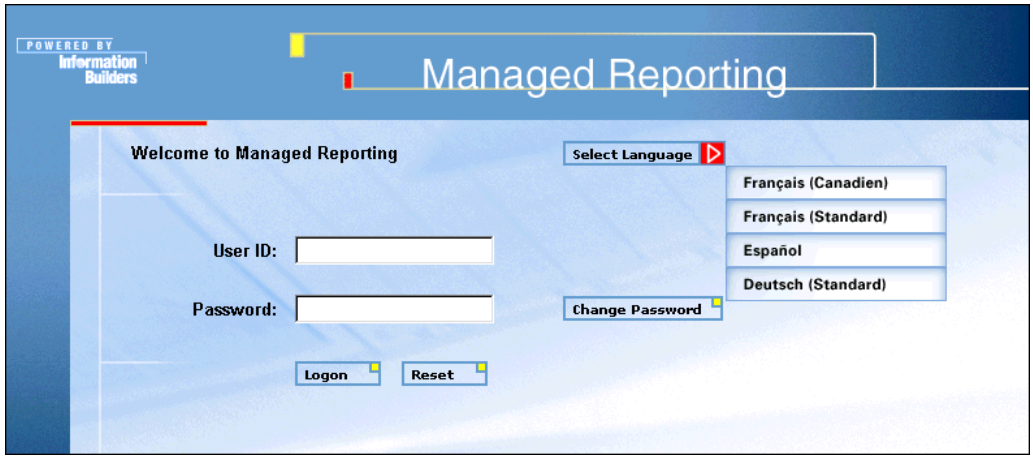
Dynamic Language Switch Feature

During installation, the Language Selection dialog box appears after you select the WebFOCUS components you want to install:

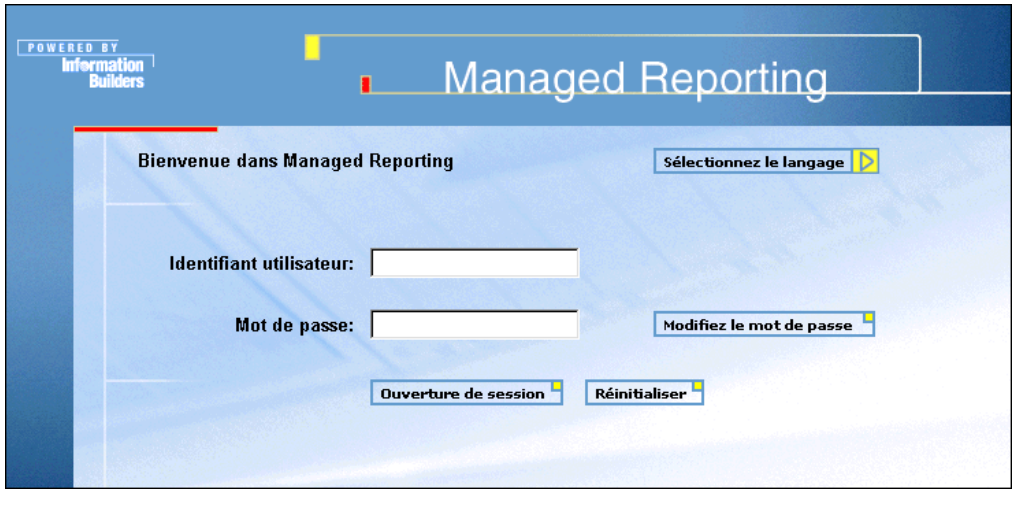


Select the languages you want to install and click *Next*. You can dynamically switch among these languages from all logon pages.

Once installation is complete, every logon page has a link with the languages installed:



The default language that displays is the operating system language specified in the regional settings or browser language options. Once you select a language, the user interface automatically switches to that language.

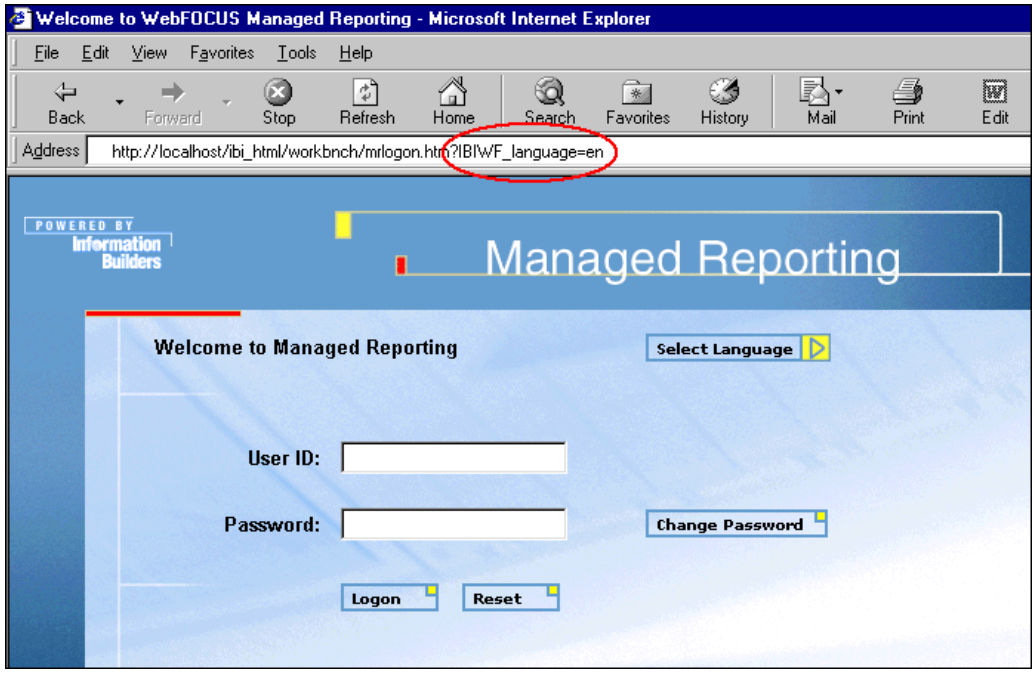


WebFOCUS creates two cookies holding the language value selected during logon. The first one is a temporary cookie, IBIWF_language. It is used to pass the language value to all WebFOCUS components during the browser session.

You may have multiple temporary cookies with different languages on the same user machine if a user has multiple browser sessions open with a different language in each one.

The second is a permanent cookie, IBIWF_langperm. It sets the initial language that is displayed every time a browser session is opened. The language may be different from the operating system's language.

For self-service applications for which a user logon is not required, you can pass the language value directly to the URL:



For example,

```
http://hostname/application_address?IBIWF_language=xx
```

where:

hostname

Is the location where WebFOCUS is installed.

application_address

Is the startup location of the application.

IBIWF_language

Is a required function call.

xx

Is the two-letter value for a valid installed language. en, for English, is the default value.

Reference: Values for Supported Languages

The following table lists the values that you can specify on the IBIWF_language function call.

Language	Value
English	en (default)
Chinese (Simplified GB)	zh
Chinese (Traditional Big-5)	tw
French (Canadian)	fc
French (Standard)	fr
German (Standard)	de
German (Austrian)	at
Japanese	ja
Korean	ko
Spanish	es

Directory Structures

Based on the languages selected during installation, the setup program creates the corresponding language folders under the ibi_html\javaassist\intl directory, which contains all the files needed to create a localized user interface. These files consist of string tables and .gif files.

Editing Language Values

The ibimultilanguage.js file, which is located in the ibi_html\javaassist\intl directory, holds the values of the languages selected in the Language Selection dialog box. For example:

```
var multiLanguage = 1;

var ArrayofLanguage = [
     ["en","English"],
     ["fc","French Canadian"],
     ["fr","French Standard"],
     ["es","Spanish"],
     ["de","German Standard"],
     ["at","German Austrian"],
     ["zh","Chinese - Simplified GB"],
     ["tw","Chinese - Traditional Big-5"],
     ["ko","Korean"],
     ["ja","Japanese"]
]
```

When you select more than one language during installation, the value of the multiLanguage variable is true (1), and therefore a language selection link is displayed on all logon pages. If you select only one language, the value is false (0); no link is present and the user interface uses one language.

All languages declared under the ArrayofLanguage variable populate the dynamic language link. The first value is the default in the event that the browser language is not available.

The administrator can make changes to the ibimultilanguage.js file.

Customizing the User Interface

In this section:

Architecture
Locating the Name of a Resource File
Interpreting the Resource File
Editing the .htm File
Example:
Locating the Name of a Resource File
Modifying Existing Text
Adding New Text
Adding New Text Without Translation
Adding a .gif File

All text that appears in a WebFOCUS window is located in a resource file. To customize the text in a window, such as a logon page, you must edit the associated resource file.

This feature allows you to add another language, provided that all resource files are translated for that language. This documentation covers customization of .htm end-user files only. For more support and information on adding a language, contact your Information Builders representative.

Architecture

Each language has its own set of resource files. Depending on the languages selected during installation, the following directory structure is available

install_drive:\ibi\WebFOCUS52\ibi_html\javaassist\intl*lang_subdir*

where:

install_drive

Is the drive on which WebFOCUS is installed.

lang_subdir

Is the subdirectory created for an installed language. The values are:

AT for German (Austrian)

DE for German (Standard)

EN for English

ES for Spanish

FC for French Canadian

FR for French Standard

JA for Japanese

KO for Korean

TW for Chinese (Traditional Big-5)

ZH for Chinese (Simplified GB)

Each subdirectory contains all the necessary files to display a localized WebFOCUS user interface, including .gif (image), .txt (text), .css (cascading style sheet), and .js (JavaScript) files.

Locating the Name of a Resource File

To make changes to the user interface in WebFOCUS, you must first locate the name of the resource file associated with the feature. Each .htm file has JavaScript code indicating its associated resource file.

Example: Locating the Name of a Resource File

This example illustrates how to locate the name of the resource file associated with the WebFOCUS Managed Reporting logon page named logon.htm.

1. Open logon.htm with a text editor such as WordPad. By default, this file resides in *install_drive*:\ibi\WebFOCUS52\ibi_html\workbnch.
2. Search for the parameter addIntlTranslatedJS in the file.

   This parameter indicates the name of the resource file associated with the .htm file.
3. Locate the name of the file that appears in double quotation marks in the JavaScript code: workbnchtrans.js.

   Workbnchtrans.js is the resource file used to populate the strings for logon.htm.

   Note: Due to formatting conventions, the line breaks shown in the code throughout this topic do not match the line breaks in the actual file.

```
.
.
.
<script type="text/javascript">addIntlTranslatedJS("workbnchtrans.js");
</script>
.
.
.
```

Interpreting the Resource File

Using the example, WebFOCUS interprets all text in \ibi\WebFOCUS52\ibi_html\workbnch\logon.htm as a single variable that must be referenced in its resource file: \ibi\WebFOCUS52\ibi_html\javaassist\intl\EN\workbnchtrans.js.

In the resource file, the first string in double quotation marks is the variable. **Do not change this variable.**

Subsequent strings in double quotation marks are the text that may appear on the logon page.

```
// $Revision:   1.xx  $
var workbnchtrans_js_Revision = "$Revision:   1.xx  $";
.
.
.
workbnchtrans.put("Select Language","Select Language");
workbnchtrans.put("Submit","Submit");
workbnchtrans.put("Reset","Reset");
workbnchtrans.put("OK","OK");
workbnchtrans.put("admin","admin");
workbnchtrans.put("Logon","Logon");
workbnchtrans.put("Logoff","Logoff");
workbnchtrans.put("Help","Help");
workbnchtrans.put("Welcome","Welcome");
workbnchtrans.put("Welcome to Managed Reporting","Welcome to Managed
Reporting");
workbnchtrans.put("Welcome to WebFOCUS Web Services","Welcome to WebFOCUS
Web Services");
workbnchtrans.put("User ID:","User ID:");
.
.
.
```

Editing the .htm File

Follow two basic rules when editing an .htm file to customize the user interface. Any new or changed text must be:

- Referenced in the applicable resource file for each language installed.

  or

- Disabled for the dynamic language switch translation through the use of the noibitranslate flag inside the preceding HTML tag.

The following are examples of editing text in an .htm file. The examples use the same files described in *Locating the Name of a Resource File* on page 6-32.

Example: Modifying Existing Text

In this example, you are going to change the labels on the Managed Reporting logon page from User ID to Customer ID, from Password to PIN, and from Change Password to Change PIN.

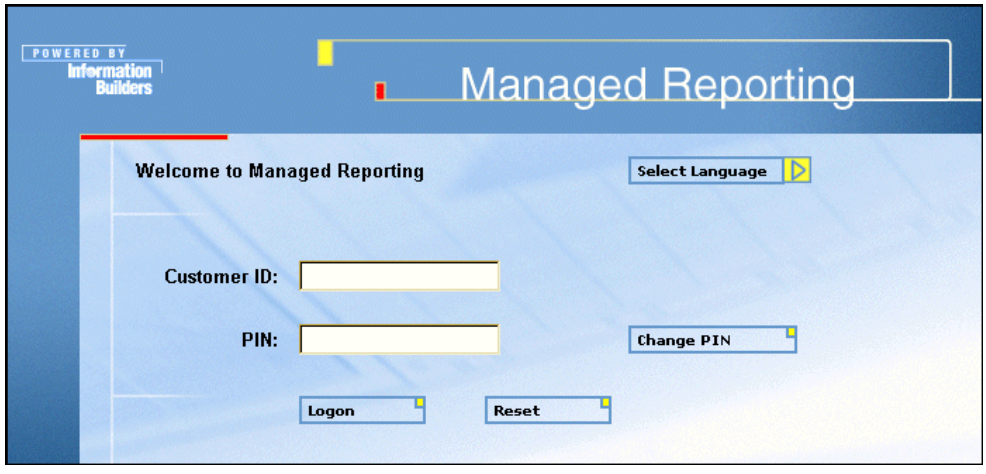
1. Open \ibi\WebFOCUS52\ibi_html\javaassist\intl\EN\workbnchtrans.js in a text editor such as WordPad, and edit the following boldfacelines to read:

```
// $Revision:   1.xx  $
var workbnchtrans_js_Revision = "$Revision:   1.xx  $";
.
.
.
workbnchtrans.put("Select Language","Select Language");
workbnchtrans.put("Submit","Submit");
workbnchtrans.put("Reset","Reset");
workbnchtrans.put("OK","OK");
workbnchtrans.put("admin","admin");
workbnchtrans.put("Logon","Logon");
workbnchtrans.put("Logoff","Logoff");
workbnchtrans.put("Help","Help");
workbnchtrans.put("Welcome","Welcome");
workbnchtrans.put("Welcome to Managed Reporting","Welcome to Managed
Reporting");
workbnchtrans.put("Welcome to WebFOCUS Web Services","Welcome to
WebFOCUS Web Services");
workbnchtrans.put("User ID:","Customer ID:");
workbnchtrans.put("Password:","PIN:");
.
.
.
workbnchtrans.put("Change Password","Change PIN");

.
.
.
}
```

2. Save and exit workbnchtrans.js.

   In a multilingual environment, repeat steps 1 and 2 for each \ibi\WebFOCUS52\ibi_html\javaassist\intl*xx**xx*workbnchtrans.js file, where *xx* is the language subdirectory or file name prefix.

3. Launch the Managed Reporting logon page. The new labels appear:



Example: Adding New Text

In this example, you want to add the following text to the Managed Reporting logon page instructing users how to log on to their account: Enter the Customer ID and PIN that were e-mailed to you.

1. Open \ibi\WebFOCUS52\ibi_html\workbnch\logon.htm in a text editor such as WordPad, and type the boldface text shown in the following example. The string *Instruction* is a variable for the actual text that will come from the resource file.

   Tip: The HTML code in your version of logon.htm may vary slightly. Make sure to place the boldface text above the HTML code that prompts for User ID.

```
.
.
.
<TD><div id="ibidropmenu"></div></TD>
.
.
.
<TR>
<TD></TD>
<TD ALIGN="left" COLSPAN="8" CLASS="tophead" >Instruction</TD>
<TD><BR><BR></TD>
</TR>
<TR>
<TD colspan="2"><IMG SRC="spacer.gif" HEIGHT=30></TD>
<TD ALIGN="right"><DIV CLASS="tophead2">User ID:</DIV></TD>
.
.
.
```

2. Save and exit logon.htm, and launch the Managed Reporting logon page:

   ```
   http://local_host/ibi_html/workbnch/mrlogon.htm
   ```

   where:

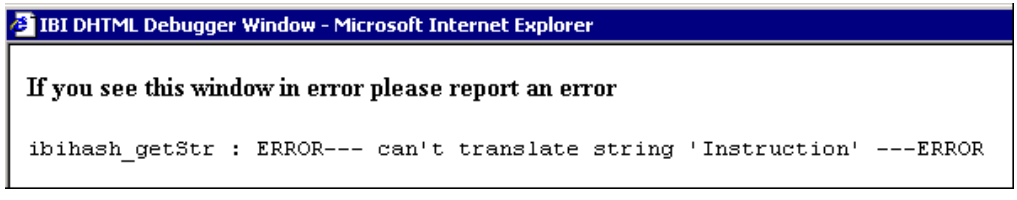
   local_host

   Is the name of the Web server on which WebFOCUS is installed.

   For now, the page displays a script message because you have not modified the resource file:

   ```
   ERROR--- can't translate string 'Instruction' ---ERROR
   ```

   Also notice the IBI DHTML Debugger Window, which displays messages that help you isolate and solve problems when customizing the user interface:

   
   IBI DHTML Debugger Window - Microsoft Internet Explorer
   If you see this window in error please report an error
   ibihash_getStr : ERROR--- can't translate string 'Instruction' ---ERROR

3. Open \ibi\WebFOCUS52\ibi_html\javaassist\intl\EN\workbnchtrans.js in a text editor such as WordPad, and add the following boldface line anywhere between the left and right brackets, { }.

   Tip: Type the new text on one line. Due to formatting conventions, the line breaks shown in the code throughout this topic do not match the line breaks in the actual file.

   ```
   // $Revision:   1.xx  $
   var workbnchtrans_js_Revision = "$Revision:   1.xx  $";
   .
   .
   .
   workbnchtrans.put("Select Language","Select Language");
   workbnchtrans.put("Submit","Submit");
   workbnchtrans.put("Instruction","Enter the Customer ID and PIN that
   were e-mailed to you.");
   .
   .
   .
   }
   ```

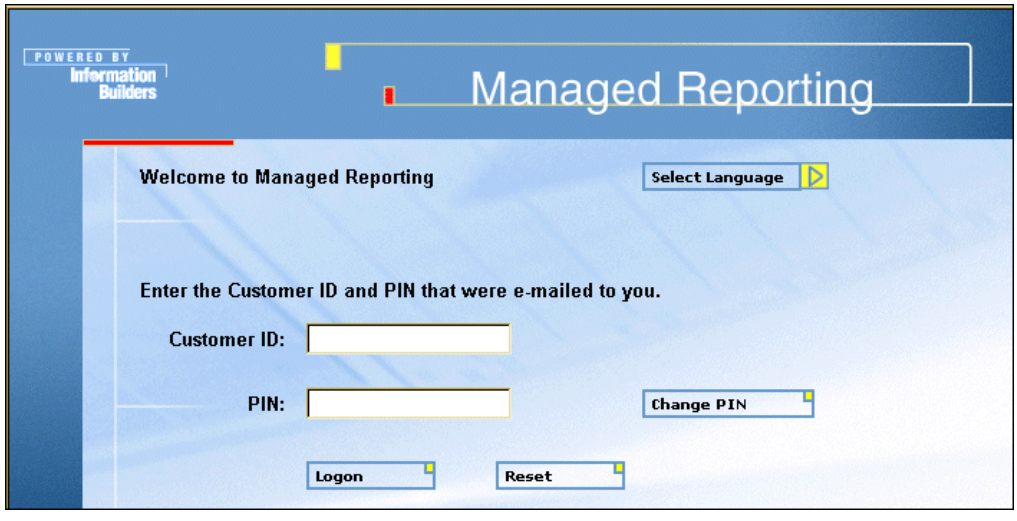
4. Save and exit workbnchtrans.js.

   In a multilingual environment, repeat steps 3 and 4 for each \ibi\WebFOCUS52\ibi_html\javaassist\intl*xx**xx*workbnchtrans.js file, where *xx* is the language subdirectory or file name prefix.

5. If you are using Microsoft Internet Explorer, clear the Temporary Internet Files cache.

   Tip: This step is not required on all machines, depending on the file storage setup.

6. Launch the Managed Reporting logon page again. The instruction you added to the page appears:



Example: Adding New Text Without Translation

You want to add text to the Managed Reporting logon page that does not require translation. In this case, use the noibitranslate flag to disable dynamic language switch translation.

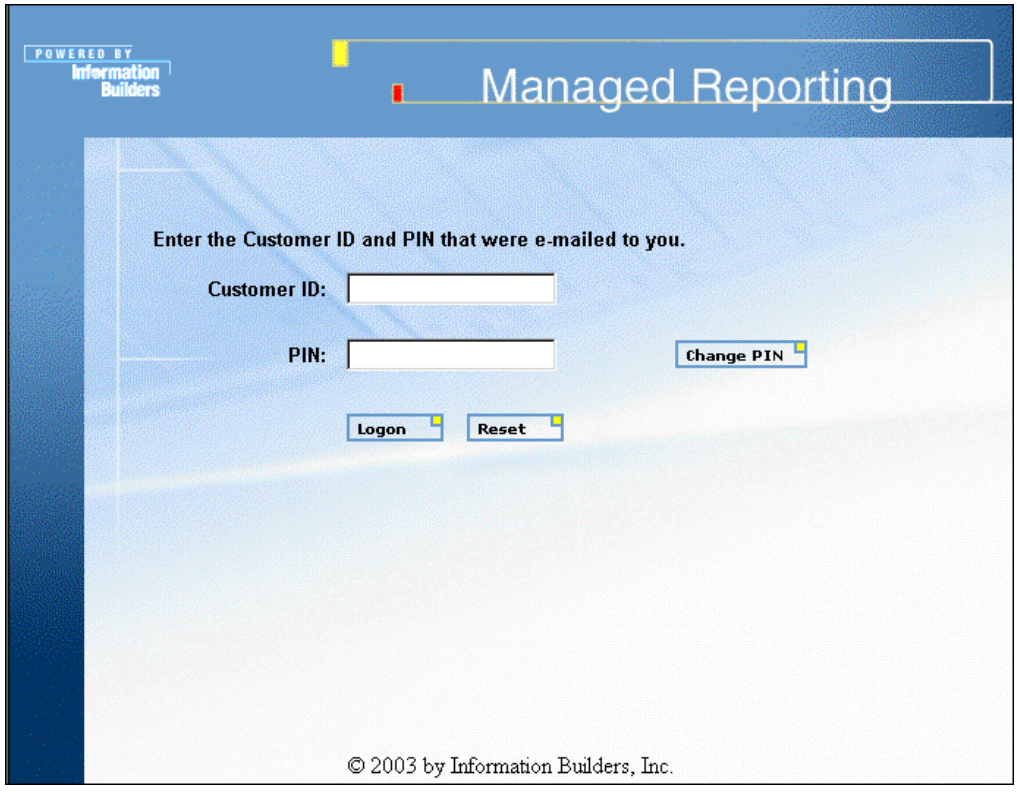
1. Open \ibi\WebFOCUS52\ibi_html\workbnch\logon.htm in a text editor such as WordPad, and add the following lines immediately before the </TABLE> tag. Notice the noibitranslate flag placement for the text © 2003 by Information Builders, Inc.:

```
.
.
.
<TR>
<TD COLSPAN="2"></TD>
<TD><BR><BR><BR><BR><BR><BR><BR><BR><BR><BR><BR><BR></TD>
<TD ALIGN="right"><IMG SRC="spacer.gif" WIDTH=1 HEIGHT=1 ALT=""></TD>
<TD COLSPAN="5" VALIGN="bottom" noibitranslate>© 2003 by Information
Builders, Inc.</TD>
<TD></TD>
<TD></TD>
</TR>
</TABLE>
.
.
.
```

2. Save and exit logon.htm.

3. Launch the Managed Reporting logon page.

The copyright text you added is near the bottom of the page. You did not need to modify the resource file workbnchtrans.js, since you used the noibitranslate flag. The same text will be displayed for all languages:



Example: Adding a .gif File

To add a red arrow to the Managed Reporting logon page, follow the same instructions in *Adding New Text* on page 6-35. Although .gif files do not require translation, you cannot use the noibitranslate flag for file names referenced in HTML code.

1. Open \ibi\WebFOCUS52\ibi_html\workbnch\logon.htm in a text editor such as WordPad, and modify the following lines that contain the *Instruction* variable to specify the path for the new .gif file:

```
.
.
.
<TD><div id="ibidropmenu"></div></TD>
.
.
.
<TR>
<TD></TD>
<TD ALIGN="left" COLSPAN="8" CLASS="tophead" VALIGN="top">
<IMG SRC="../javaassist/arrow.gif" WIDTH=15 HEIGHT=15>Instruction</TD>
<TD><BR><BR></TD>
</TR>
.
.
.
```

   Tip: This example is a continuation of the previous examples, so you will see the line added previously to supply text for the variable *Instruction*.

2. Save and exit logon.htm.

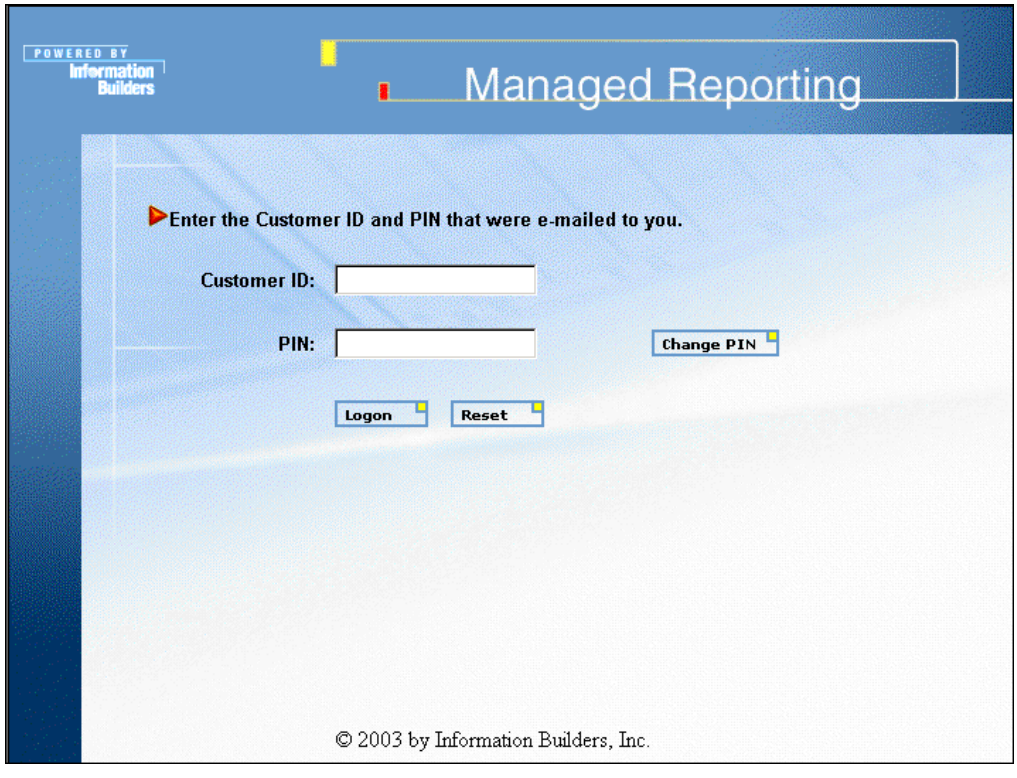
3. Open \ibi\WebFOCUS52\ibi_html\javaassist\intl\EN\workbnchtrans.js in a text editor such as WordPad, and add the following boldface line as shown:

```
// $Revision:   1.xx  $
var workbnchtrans_js_Revision = "$Revision:   1.xx  $";
.
.
.
workbnchtrans.put("Select Language","Select Language");
workbnchtrans.put("Submit","Submit");
workbnchtrans.put("Instruction","Enter the Customer ID and PIN that
were e-mailed to you.");
workbnchtrans.put("arrow.gif","/ibi_html/javaassist/arrow.gif");
.
.
.
}
```

4. Save and exit workbnchtrans.js.

   In a multilingual environment, repeat steps 3 and 4 for each \ibi\WebFOCUS52\ibi_html\javaassist\intl*xx**xx*workbnchtrans.js file, where *xx* is the language subdirectory or file name prefix.

5. Launch the Managed Reporting logon page. The red arrow you added appears.



CHAPTER 7

WebFOCUS Reporting Server Enhancements

This chapter describes all of the features for the WebFOCUS Reporting Server in Version 5 Release 2. Topics that contain Version 5 Release 2.3 features are marked with asterisks.

For complete documentation about server configuration and administration, see the *iWay Server Administration for UNIX, Windows, OpenVMS, OS/400, OS/390 and z/OS* manual.

Topics:

Remote Installation and Control

* Installing WebFOCUS Reporting Server Remotely
* Controlling WebFOCUS Reporting Server Remotely

Server Administration

- Agent Services
- Blocking New Connections on a Running Server
- Enhanced Connection Management
- Specifying the Location of User Profiles
* Configuring National Language Support
* Specifying the Number of Lines in the EDAPRINT Log
* Automatic Self Recovery
- Tree View for Procedures
- Text Console Enhancements
- Communicating With a FOCUS Database Server

Server Security

- Windows Authentication Methods
- Assigning Administrative Privileges
- Protecting the WebFOCUS Reporting Server Console
- Windows Trusted Client
- HTTP Basic Authentication
- SSL/TLS Support
* Lawson Security Adapter
* C2 Security Support

OS/390 and z/OS Server

- Data Adapter Support
- Creating Synonyms
- Accessing Files on z/OS and OS/390
- External Sort
- Hiper Memory
- Accessing z/OS and OS/390 Applications From the WebFOCUS Reporting Server Console
- Accessing z/OS and OS/390 VSAM Files Using XMI Server
- Security Types
- Server Accounting Using SMF Records
- Workload Manager Enclave Integration

Platform Support

- Support for Additional Operating Systems and Releases
- Support for Large Files on HP and Solaris

Listeners

- Displaying Listener Statistics in the WebFOCUS Reporting Server Console
- MQ XML Listener
- SOAP XML Listener

Installing WebFOCUS Reporting Server Remotely

In Release 5.2.3, remote installations reduce the manual tasks involved in installing a server on UNIX, OS/390, z/OS, OS/400, and OpenVMS. Server installation is simplified because:

- The installation files can be executed and transferred from a Windows machine.
- Locations and naming conventions are set rather than variable.

Remote installation involves the following tasks:

1. Prepare your *target* computer (the UNIX, OS/390, z/OS, OS/400, or OpenVMS computer where installing the server), ensuring that it meets the server installation requirements.
2. Download the installation files from the iWay Web site.
3. Install the iWay Client on a Windows machine.
4. Install the server on the target computer using the iWay Client.
5. Test the installation of your server.

Controlling WebFOCUS Reporting Server Remotely

In Release 5.2.3, after the installation is complete, you can use the iWay Client Web Console to remotely control and access the server.

In addition, you can use this Console as a single point of control for an entire network of servers. To install additional servers, you can expand the *Add* folder on the Remote Servers page, choose how to connect to the server, and define a node for the server.

Agent Services

You can divide agents into different groups called services. Each agent runs for a specific service. Each service can have different values for some of the configuration parameters.

Services are configured in the server configuration file, edaserve.cfg. At least two services must be configured in edaserve.cfg:

- **DEFAULT.**
- **WC_DEFAULT.** This service is used only for administrative requests from the WebFOCUS Reporting Server Console.

More services can be configured as required.

To manually define a service in edaserve.cfg, use the following format:

```
SERVICE = <service_name>
BEGIN
 deployment  =  <private|pooled>
 eda_user_fixed  =  <yes|no>
 pooled_user  =  <pooled_user_id>
 pooled_password  =  <pooled_user_password>
 maximum  =  <number>
 number_ready  =  <number>
 maximum_q  =  <number>
 queue_limit  =  <number>
 idle_session_limit  =  <number>
 idle_agent_limit  =  <number>
 cpu_limit  =  <number>
 memory_limit  =  <number>
 max_sessions_per_agent  =  <number>
 profile  =  <file.fex>
 max_connections_per_user  =  <number>
END
```

Note that there are two additional keywords in the service block:

`profile`

Specifies the name of a profile to be executed during startup of agents for this service. By default, the baseapp directory is searched for the profile. To specify a file in another application directory, use the following format: APP/file.fex.

`max_connections_per_user`

Specifies the maximum number of concurrent connections a user can have. If this parameter is not set or is set to -1, the number of concurrent connections is unlimited; this is the default behavior.

All other parameters have the same meanings and relationships as they had for the entire server, applied now in the scope of a service.

To connect to an agent of a specific service, use the following format in the communication configuration file, odin.cfg, for the client:

```
; TCP Loopback Client
NODE = LOOPBACK
BEGIN
 PROTOCOL = ...
 CLASS = CLIENT(<service_name>)
 HOST = ...
 SERVICE = ...
END
```

If no service name is specified, by default the client is connected to an agent from the DEFAULT service.

Blocking New Connections on a Running Server

To facilitate application changes on a running production server, a control has been introduced to allow an administrator to block new connections from non-administrative users. You can access this feature as follows:

- From the WebFOCUS Reporting Server Console, it is available in the Workspace submenu as Quiesce or Enable.
- From the text console, to disable new connections, use -quiesce newconnect. To enable again, use -enable newconnect.

When a server is quiesced, new connections from a Server Administrator can still go through because some configuration changes from the WebFOCUS Reporting Server Console require an agent connection. New connections from non-administrators are rejected with a resource limit error (-33).

The administrator must continue to monitor existing sessions, wait for them to finish before disrupting (for example, stopping old agents or restarting the server after configuration changes), and/or manually terminate selected sessions.

The quiesced/enabled state is also preserved in the server configuration file, edaserve.cfg, under the keyword quiesce_connections so that the server can stay quiesced across one or more restarts during configuration changes.

Enhanced Connection Management

The connections view in the WebFOCUS Reporting Server Console has been enhanced to show both active and queued connections. In previous releases, only queued connections were represented on the console. The queue view has been removed.

You can see the statistics of any connection, either active or queued, from the Connections Panel on the WebFOCUS Reporting Server Console. You can also cancel the connection from this panel.

Specifying the Location of User Profiles

You can specify the location of user profiles. A user profiles directory is created during installation and pointed to by a new keyword, edaprfu, in the server configuration file, edaserve.cfg. This user profiles directory is located, by default, under the ibi directory (for example, in a Windows environment, c:\ibi\profiles).

Overwrite the default location of the user profiles directory by setting an environment variable, EDAPRFU, or by assigning the location to the edaprfu keyword in edaserve.cfg. When a user connects to the server, the location pointed to by the environment variable EDAPRFU is searched. If this variable is not set or the user profile is not found in the specified directories, then the location pointed to by the edaprfu keyword in edaserve.cfg is searched.

Configuring National Language Support

In this section:

Step 1. Determine Which Code Page You Need

Step 2. Configure the WebFOCUS Reporting Server for NLS

Step 3. Configure the WebFOCUS Client for NLS

Configuring the Server and Client for Baltic Code Pages

How to:

Manually Specify the Language and Code Page Values

Edit the NLS Configuration File (nlscfg.err)

Configure the Server and Client for Baltic Code Pages on Windows

Reference:

Default Code Page Generation File (Windows, UNIX, OS/400)

Supported Code Page File (Windows, UNIX, OS/400)

New NLS Configuration File (nlscfg.err) Parameters

Language and Default Code Pages for Windows and UNIX

Language and Default Code Pages for OS/400

You can use the NLS (National Language Support) Configuration Wizard to select the code page for the data sources your server will access. Access the NLS Configuration Wizard through the WeFOCUS Reporting Server Console.

A code page is the computer representation of a character set. Each written symbol in a language is assigned a unique number, usually expressed in hexadecimal notation. A code page has a unique identification number to distinguish the operating system and language or languages to which it applies.

The WebFOCUS Reporting Server maintains a default code page generation file. After you configure the server using the WebFOCUS Reporting Server Console, WebFOCUS updates the default file with the code page selected for the server and with the associated client code pages. The updated file, which contains the new code page values and the default values, is used to generate the transcoding, monocasing, and sorting tables. For the default code page values, see *Default Code Page Generation File (Windows, UNIX, OS/400)* on page 7-6.

To configure the server for NLS, perform the following steps:

Step 1. Determine Which Code Page You Need

Step 2. Configure the WebFOCUS Reporting Server for NLS

Step 3. Configure the WebFOCUS Client for NLS

Step 1. Determine Which Code Page You Need

Today, most Database Management Systems (DBMSs) have ways to refer to the code page associated with the data. See your database administrator to get this information so that you can select the corresponding server code pages to interpret the data correctly.

By default, the server uses code page 437 (AMENGLISH) on Windows and UNIX, and code page 37 (AMENGLISH) on OS/400, OS/390, and z/OS. For instructions on configuring the server and client on OS/390 and z/OS, see the documentation for those platforms in the *National Language Support for International Computing* manual.

To identify supported code pages, refer to the file cpxcptbl.nls, which resides in the nls directory under the installation directory (for example, on Windows, \ibi\srv52\home\nls).

If the code page needed to interpret your data is not listed in the supported code page file, cpxcptbl.nls, contact your Information Builders representative.

Reference: Default Code Page Generation File (Windows, UNIX, OS/400)

The following is the default code page generation file, cpcodepg.nls, for Windows, UNIX, and OS/400.

```
CP00037   E SBCS  US IBM MF EBCDIC code
CP00437   A SBCS  US PC ASCII code
CP00137   A SBCS  ANSI Character Set for MS-Windows
CP01047   E SBCS  IBM MF Open Systems (Latin 1)
CP65001   A UTF8  Unicode (UTF-8)
```

Reference: Supported Code Page File (Windows, UNIX, OS/400)

The following is the supported code page file, cpxcptbl.nls, for Windows, UNIX, and OS/400. In Release 5.2.3, WebFOCUS added support for the three Baltic code pages at the end of the file. For instructions on configuring the server and client for these code pages, see *Configuring the Server and Client for Baltic Code Pages* on page 7-15.

```
* The active Code Page information is represented by lines whose first 2
* columns are 'CP'.
* Valid DBCSID : SOSI,SJIS,EUC,KPC,KKS,TPC,TBIG5,SOSIF,SOSIH
*   C - Character type : E = EBCDIC
*                        A = ASCII
*   DBCS - DBCS ID     : SBCS = Single Byte Character Set
*                        SOSI = IBM MF
*                        SJIS,EUC,SOSIF,SOSIH = Japanese
*                        KPC,KKS = Korean
*                        TPC,TBIG5,TNS,TTEL = Taiwanese
*   note: NRC Set = National Replacement Character Set
*         MF      = Mainframe
*
Codepage  C DBCS  Description
--------  - ----  ------------------------------------------------------
CP00037   E SBCS  US IBM MF EBCDIC code
CP00437   A SBCS  US PC ASCII code
CP00137   A SBCS  ANSI character set for MS-Windows
CP00500   E SBCS  IBM MF International European
CP00273   E SBCS  IBM MF Germany F.R./Austria
CP00277   E SBCS  IBM MF Denmark, Norway
CP00278   E SBCS  IBM MF Finland, Sweden
CP00280   E SBCS  IBM MF Italy
CP00281   E SBCS  IBM MF Japanese English
CP00284   E SBCS  IBM MF Spain/Latin America
CP00285   E SBCS  IBM MF United Kingdom
CP00297   E SBCS  IBM MF France
CP00420   E SBCS  IBM MF Arabic Bilingual
CP00424   E EBCH  IBM MF Israel(Hebrew)
CP00850   A SBCS  IBM PC Multinational
CP00856   A SBCS  IBM PC Hebrew
CP00860   A SBCS  IBM PC Portugal
CP00861   A SBCS  IBM PC Iceland
CP00862   A ASCH  IBM PC Israel
CP00863   A SBCS  IBM PC Canadian French
CP00864   A SBCS  IBM PC Arabic
CP00865   A SBCS  IBM PC Nordic
CP00869   A ASCG  IBM PC Greece
CP00871   E SBCS  IBM MF Iceland
CP00875   E EBCG  IBM MF Greece
CP00637   A SBCS  DEC Multinational Character Set
CP00600   A SBCS  DEC German NRC Set
CP00604   A SBCS  DEC British ( United Kingdom ) NRC Set
```

```
CP00608   A SBCS  DEC Dutch ( Netherland ) NRC Set
CP00612   A SBCS  DEC Finnish ( Finland ) NRC Set
CP00616   A SBCS  DEC French ( Flemish and French/Belgian ) NRC Set
CP00620   A SBCS  DEC French Canadian NRC Set
CP00624   A SBCS  DEC Italian ( Italy ) NRC Set
CP00628   A SBCS  DEC Norwegian/Danish ( Norway,Denmark ) NRC Set
CP00632   A SBCS  DEC Portugal NRC Set
CP00636   A SBCS  DEC Spanish ( Spain ) NRC Set
CP00640   A SBCS  DEC Swedish ( Sweden ) NRC Set
CP00644   A SBCS  DEC Swiss ( Swiss/French and Swiss/German ) NRC Set
CP00930   E SOSI  Japanese IBM MF Katakana Code Page (cp290+cp300)
CP00939   E SOSI  Japanese IBM MF Latin Extended (cp1027+cp300)
CP00932   A SJIS  Japanese PC Shift-JIS (cp897+cp301)
CP00942   A SJIS  Japanese PC Shift-JIS Extended (cp1041+cp301)
CP10942   A EUC   Japanese PC EUC
CP10930   E SOSIF Japanese Mainframe ( Fujitsu )
CP20930   E SOSIH Japanese Mainframe ( Hitachi )
CP00933   E SOSI  Korean IBM MF Extended (cp833+cp834)
CP00934   A KPC   Korean IBM PC (cp891+cp926)
CP00944   A KPC   Korean IBM PC Extended (cp1040+cp926)
CP00949   A KKS   Korean KS5601 code (cp1088+cp951)
CP00935   E SOSI  PRC IBM MF Extended (cp836+cp837)
CP00946   A CPC   PRC IBM PC Extended (cp1042+cp928)
CP00937   E SOSI  Taiwanese IBM MF (cp37+cp835)
CP00948   A TPC   Taiwanese IBM PC Extended (cp1043+cp927)
CP10948   A TBIG5 Taiwanese PC BIG-5 code (cp950 ?)
CP20948   A TNS   Taiwanese PC National Standard code
CP30948   A TTEL  Taiwanese PC Telephone code
CP00857   A ASCR  IBM PC Turkish (Latin 5)
CP00920   A ISO9  ISO-8859-9 (Latin 5, Turkish)
CP01026   E EBCR  IBM MF Turkish
CP01252   A SBCS  Windows, Latin 1
CP00819   A SBCS  ISO-8859-1 (Latin 1, IBM Version 8X/9X undef'ed)
CP00912   A ISO2  ISO-8859-2 (Latin 2)
CP01047   E SBCS  IBM MF Open Systems (Latin 1)
CP65001   A UTF8  Unicode (UTF-8)
CP00916   A ISO8  ISO-8859-8 (Hebrew)
CP00813   A ISO7  ISO-8859-7 (Greek)
CP00870   E EBCE  IBM MF Multilingual (Latin 2, East European)
CP00852   A ASCE  IBM PC Multilingual (Latin 2, East European)
CP01250   A WINE  Windows, Latin 2 (East European)
CP01253   A WING  Windows, Greek
CP01254   A WINR  Windows, Turkish
CP01255   A WINH  Windows, Hebrew
CP01025   E EBCY  IBM MF Cyrillic, Multilingual
CP00866   A ASCY  IBM PC Cyrillic #2
CP00915   A ISO5  ISO-8859-5 (Cyrillic, 8-bit)
CP01251   A WINY  Windows, Cyrillic
CP01112   E EBCB  IBM MF Baltic Multilanguage
CP00921   A ISO3  ISO-8859-4 (Baltic Multilanguage)
CP01257   A WINB  Windows, Baltic (Latin 7)
```

Step 2. Configure the WebFOCUS Reporting Server for NLS

1. Open the WebFOCUS Reporting Server Console with administrative privileges. In a Web browser, enter the following address

   `http://host:port`

   where:

   host

   Is the host name or IP address of the machine running the WebFOCUS Reporting Server.

   port

   Is one number higher than the port number specified when installing the server. For example, if at installation you accepted the default port number of 8120, to access the console you would specify port number 8121.

   If you installed the server on a machine with the host name MyServer and chose the base port 9190, to access the console enter http://MyServer:9191.

   In WebFOCUS running on Windows, you can also access the console in the following ways:

   - Select *WebFOCUS 52 Server* and then *Web Console* from the Start Programs menu.
   - In Developer Studio, select *Server Console* from the Command menu. The console for the project-based development environment opens.

2. In the console's navigation pane, click *NLS*.

   The NLS Configuration Wizard opens. Notice that the current settings near the top of the window show the default configuration for the code page, code page name, and language.

   The following steps describe how to select the code page using graphical prompts. Alternatively, you can manually specify the language and code page values. For instructions, see *How to Manually Specify the Language and Code Page Values* on page 7-10.

   Use the manual procedure to configure the server for a code page that is different from the default associated with a language, or to configure the server for features such as currency symbol or Microsoft Excel language.You may also use the manual procedure if you already know the language and default code page value.

3. To select a standard ISO or DBCS code page, choose it from the scroll-down list.

   To select a code page that predates the introduction of Windows technology, first select the *Legacy OEM (DOS) Code Pages* radio button, and then choose the page from the scroll-down list. These pages are not appropriate for current Windows applications; they are supplied for backward compatibility.

4. If you select *Western European (ISO 8859-1)*, or the legacy code page *Western European (850)* or *Nordic (865)*, you are prompted for the language of the data source. This selection controls the language of server error messages and other internal locale capabilities.

   Click the desired language from the list box on the right.

   An asterisk to the right of a language—for example, French*—means that translated server error messages (FOC*nnn* through FOC*nnnnn*) are available and the WebFOCUS Reporting Server Console is fully displayed in that language once you configure and restart the server.

   If you choose *Other*, or a language without an asterisk, the server messages and console are displayed in English.

5. Click *Configure and Restart*.

   A message indicates that the server is restarting. At this point, the server is creating the necessary transcoding, sorting, and monocasing tables for the selected code page.

Procedure: How to Manually Specify the Language and Code Page Values

1. From the NLS Configuration Wizard, click *Advanced NLS Configuration*. The advanced NLS Configuration window opens, showing the current values for the language (LANG) and code page (CODE_PAGE).
2. In the LANG field, enter either the language name or language abbreviation. See *Language and Default Code Pages for Windows and UNIX* on page 7-12, or *Language and Default Code Pages for OS/400* on page 7-13 for values. To specify the language of server error messages, enter one of the following: AMENGLISH, DUTCH, FRENCH, GERMAN, JAPANESE, or SPANISH.
3. In the CODE_PAGE field, enter a code page based on the data sources that the server will access. You may enter a code page that is different from the default associated with a language.

4. Click *Save and Restart*.

   A message indicates that the server is restarting. At this point, the server is creating the necessary transcoding, sorting, and monocasing tables for the code page selected.

   You can configure the server for other features by editing the NLS configuration file (nlscfg.err), which controls settings such as currency symbol, date and time format, and Microsoft Excel language.

Procedure: How to Edit the NLS Configuration File (nlscfg.err)

The NLS configuration file (nlscfg.err) controls the value of the code page and the language of server error messages. It is located in the etc directory under the configuration directory, for example, on Windows, \ibi\srv52\wfs\etc.

1. From the advanced NLS Configuration window, click *Edit nlscfg.err*. The content of the current nlscfg.err file opens in an editor.
2. Enter the desired parameters. See *New NLS Configuration File (nlscfg.err) Parameters* on page 7-11 for new settings you can control.

Reference: New NLS Configuration File (nlscfg.err) Parameters

`EXL2KLANG =`

Is the language used on the WebFOCUS Reporting Server for Microsoft Excel requests. It must be the same as the language of Excel on the machine running the browser to ensure the correct display of output from the WebFOCUS Reporting Server. Possible values are:

`ENG` for English. ENG is the default value.

`FRE` for French.

`GER` for German.

`JPN` for Japanese.

`KOR` for Korean.

`SPA` for Spanish.

`EXL2K_DEFCOLW =`

Is the column width for the language of Excel. It tells WebFOCUS the value to use for Excel formatting. For English or Japanese, the default column width is 8. Western European languages use the default column width of 10.

You must supply the Excel column width that is valid for the language specified on EXL2KLANG. It is a fixed value for each language.

Step 3. Configure the WebFOCUS Client for NLS

To configure the WebFOCUS Client for NLS, see *Configuring WebFOCUS Client Settings* in Chapter 6, *WebFOCUS Client Enhancements*.

Reference: Language and Default Code Pages for Windows and UNIX

In Release 5.2.3, WebFOCUS added support for the Baltic language.

Language Code	Language Name	Language Abbreviation	Code Page
001	AMENGLISH	AME	437
001	ENGLISH	ENG	437
370	BALTIC	BAL	921
420	CZECH	CZE	912
045	DANISH	DAN	137
031	DUTCH	DUT	137
358	FINNISH	FIN	137
033	FRENCH	FRE	137
049	GERMAN	GER	137
030	GREEK	GRE	813
972	HEBREW	HEB	916
972	HEBREW	HEW	1255
039	ITALIAN	ITA	137
081	JAPANESE	JPN	942
081	JAPANESE	JPE	10942
082	KOREAN	KOR	949
047	NORWEGIAN	NOR	137
048	POLISH	POL	912
351	PORTUGUESE	POR	137

Language Code	Language Name	Language Abbreviation	Code Page
007	RUSSIAN	RUS	915
085	S-CHINESE	PRC	946
034	SPANISH	SPA	137
046	SWEDISH	SWE	137
086	T-CHINESE	ROC	10948
090	TURKISH	TUR	920
044	UKENGLISH	UKE	137
800	UNICODE	UCS	65001

Reference: Language and Default Code Pages for OS/400

In Release 5.2.3, WebFOCUS added support for the Baltic language.

Language Code	Language Name	Language Abbreviation	Code Page
001	AMENGLISH	AME	37
001	ENGLISH	ENG	37
370	BALTIC	BAL	1112
420	CZECH	CZE	870
045	DANISH	DAN	277
031	DUTCH	DUT	37
358	FINNISH	FIN	278
033	FRENCH	FRE	297
049	GERMAN	GER	273
049	GERMAN-500	GE5	500
030	GREEK	GRE	875
972	HEBREW	HEB	424

Language Code	Language Name	Language Abbreviation	Code Page
039	ITALIAN	ITA	280
081	JAPANESE	JPN	939
081	JAPANESE-930	JPK	930
082	KOREAN	KOR	933
047	NORWEGIAN	NOR	277
048	POLISH	POL	870
351	PORTUGUESE	POR	37
007	RUSSIAN	RUS	1025
085	S-CHINESE	PRC	935
034	SPANISH	SPA	284
046	SWEDISH	SWE	278
086	T-CHINESE	ROC	937
090	TURKISH	TUR	1026
044	UKENGLISH	UKE	285

Configuring the Server and Client for Baltic Code Pages

In Release 5.2.3, WebFOCUS added support for Baltic code pages 1112, 921, and 1257. You must manually configure the WebFOCUS Reporting Server and WebFOCUS Client for these code pages.

Procedure: How to Configure the Server and Client for Baltic Code Pages on Windows

1. Open the file cpcodepg.nls in the directory \ibi\srv52\wfs\etc. Add the following three lines to the file and save it:

   ```
   CP01112 - IBM MF Baltic Multilanguage
   CP00921 - ISO-8859-4 (Baltic Multilanguage)
   CP01257 - Windows, Baltic (Latin 7)
   ```

2. Use the advanced NLS configuration option to manually enter the LANG and CODE_PAGE values from the WebFOCUS Reporting Server Console, as described in *How to Manually Specify the Language and Code Page Values* on page 7-10.

   On the NLS Configuration window, specify

   ```
   LANG = BALTIC
   CODE_PAGE = CP0nnnn
   ```

   where:

   nnnn

   Is the Baltic code page for the data sources.

3. Generate the transcoding tables by clicking *Save and Restart*.

4. To configure the WebFOCUS Client, use the User Defined option to manually enter the code page, as described in *Configuring WebFOCUS Client Settings* in Chapter 6, *WebFOCUS Client Enhancements*.

Specifying the Number of Lines in the EDAPRINT Log

How to:

Specify the Maximum Number of Lines in a Server Activity Log

Reference:

Identifying Log Continuation Files

In Release 5.2.3, the configuration parameter edaprint_max_lines specifies the maximum number of lines that the server writes to a single server activity log (edaprint) file. When the specified number is reached, the current log file is archived and a new log file is created.

Note: The lines generated by the server start process are not included in the count. As a result, the first log file in the session contains more lines than the specified maximum. It is larger by the number of lines in the startup information.

Syntax: How to Specify the Maximum Number of Lines in a Server Activity Log

```
edaprint_max_lines = {0|maximum_number_of lines}
```

where:

maximum_number_of lines

Is any number greater than 0. 0 is the default value, which indicates no limit.

The recommended value is 1000. This value ensures that, even on a fast system, a log file remains active for at least one second, which is long enough to prevent two log files having the same timestamp. The timestamp determines the order in which log files are deleted, so it is critical that no two files have the same timestamp.

Reference: Identifying Log Continuation Files

When a new log file is created, the first line of the new file indicates the file's order in the log archive and the name of the first archived file in the series as follows

```
continuation #n of edapriNN
```

where:

n

Is the file's rank in the log file history.

edpri*NN*

Indicates the startup file in the series. The value of *NN* is automatically assigned by the server when the log file is archived.

Automatic Self Recovery

In Release 5.2.3, the server provides the ability to automatically self-recover crashed listeners and special services without restarting the server. You can enable/disable this functionality using the automatic_recovery keyword in the server configuration file, edaserve.cfg.

Tree View for Procedures

The Procedures page of the WebFOCUS Reporting Server Console has been enhanced to support a directory tree structure. Directories on the Application Path are displayed in a tree format with procedures or HTML files appearing in individual folders. Clicking a procedure displays the options that can be performed with that procedure. Selecting an HTML file from the Procedures page opens that file.

Text Console Enhancements

Two options have been added to the edastart command, which is used to start the Workspace Manager from the command line and to invoke the Text Console.

`-quit`

Enables you to exit the Text Console and return to the shell without stopping the Workspace Manager. Using this option suspends the display of edaprint log messages in the Text Console, but does not interrupt the flow of these messages to the edaprint log itself.

`-console`

Enables you to return to the Text Console and resume monitoring of subsequent edaprint messages from the console. This option invokes the Text Console even if it was not previously running (that is, if the Workspace Manager was started using the -start option, which does not invoke a Text Console).

You cannot use this option if the Text Console is already running, as there can be only one Text Console.

For information on edastart options, issue the following command:

```
edastart -?
```

Communicating With a FOCUS Database Server

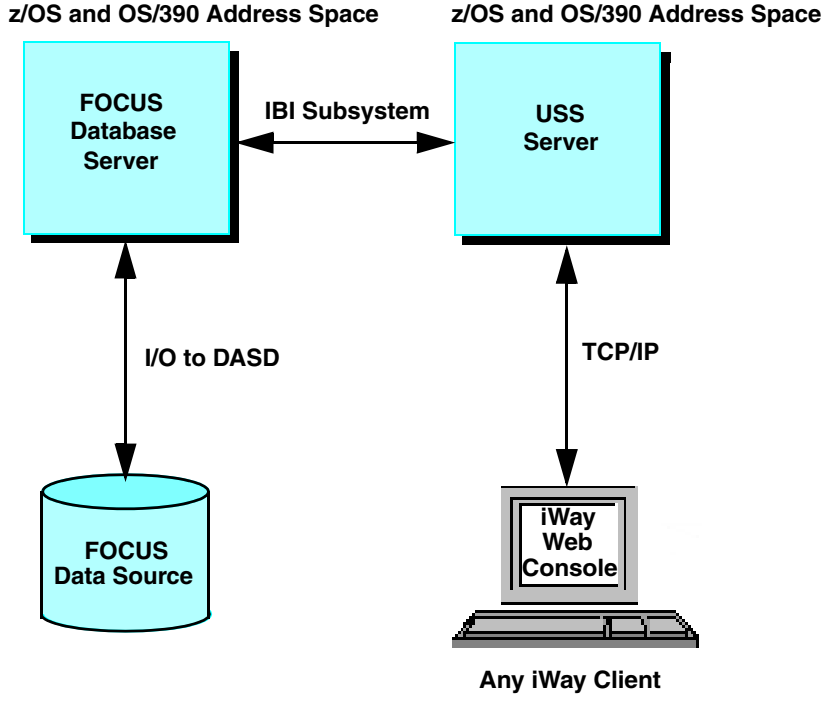
How to:

Access a FOCUS Database Server on z/OS and OS/390

Example:

Retrieving Data From a FOCUS Database Server on z/OS and OS/390

The WebFOCUS Reporting Server running under UNIX Systems Services (USS) can communicate with one or more FOCUS Database Servers (sink machines) running in the z/OS and OS/390 environment, in addition to one FOCUS Database Server running under the WebFOCUS Reporting Server for OS/390 and z/OS.



This feature is supported in OS/390 2.10 and higher operating systems. It is not supported on VM operating systems.

Syntax: How to Access a FOCUS Database Server on z/OS and OS/390

To enable communication between the Server for OS/390 and z/OS and a FOCUS Database Server on z/OS and OS/390, perform the following steps:

1. In the communications configuration file, odin.cfg, of the OS/390 and z/OS, add an SBS node block

```
; FOCUS Database Client
NODE = sbsname
BEGIN
 PROTOCOL = SBS
 CLASS = SUCLIENT
 SERVICE = qualif.focsu.data
END
```

   where:

   sbsname

   Is a unique node name. This is also the ddname for the FOCUS Database Server communication data set. The server automatically performs the allocation of the communication data set based on the information in this configuration block. (The FOCUS Database Server startup JCL must allocate the same data set to ddname FOCSU.) This is the SU name that must be specified in the USE command needed to access a data source on the FOCUS Database Server.

   Note: During installation of the Server for OS/390 and z/OS, the odin.cfg file is created with two blocks named FOCSU and FOCSU01. These blocks provide access to the FOCUS Database Server running under the Server for OS/390 and z/OS. You cannot use either of these two names for this new block.

   qualif.focsu.data

   Is the fully qualified name of the z/OS and OS/390 FOCUS Database Server communication data set.

2. For each FOCUS data source to be accessed on the FOCUS Database Server, issue a USE command in any supported profile or in a procedure on the Server for OS/390 and z/OS:

```
USE focusdb1 ON sbsname
USE focusdb2 ON sbsname
```

Example: Retrieving Data From a FOCUS Database Server on z/OS and OS/390

The odin.cfg file on the Server for OS/390 and z/OS contains the following blocks. The first two are created automatically. The third block has been added for communicating to a FOCUS Database Server on z/OS and OS/390:

```
; FOCUS Database Server
NODE = FOCSU
BEGIN
 PROTOCOL = TCP
 CLASS = SUSERVER
 SERVICE = 3562
END

; FOCUS Database Client
NODE = FOCSU01
BEGIN
 PROTOCOL = TCP
 CLASS = SUCLIENT
 HOST = localhost
 SERVICE = 3562
END

; FOCUS Database Client against z/OS and OS/390 Sink Machine
NODE = FOCSBS
BEGIN
 PROTOCOL = SBS
 CLASS = SUCLIENT
 SERVICE = USER1.FOCSU.DATA
END
```

The following REQUEST has a USE command that directs retrieval to the FOCUS Database Server on z/OS and OS/390.

```
USE GGSALES ON FOCSBS
END
SET PAGE = NOPAGE
TABLE FILE GGSALES
 SUM UNITS DOLLARS
 BY REGION BY CATEGORY
END
```

The output is:

```
Region      Category    Unit Sales   Dollar Sales
------      ---------   ----------   ------------
Midwest     Coffee      332777       4178513
            Food        341414       4338271
            Gifts       230854       2883881
Northeast   Coffee      335778       4164017
            Food        353368       4380004
            Gifts       227529       2848289
Southeast   Coffee      350948       4415408
            Food        349829       4308731
            Gifts       234455       2986240
West        Coffee      356763       4473527
            Food        340234       4202338
            Gifts       235042       2977092
```

Windows Authentication Methods

Security on Windows NT/2000 has been enhanced to support different Microsoft authentication methods. This is done with a keyword in the server configuration file, edaserve.cfg. These methods correspond to a user's privileges within Microsoft security. A method indicates the mode used to authenticate and impersonate the client at the operating system level. Windows allows access only for users that have the corresponding user right.

```
logon_method = interactive [/batch /network]
```

`interactive`

Corresponds to the user privilege *log on locally*. interactive is the default value.

`network`

Corresponds to the user privilege *access this computer from the network*.

`batch`

Corresponds to the user privilege *log on as a batch job*.

For example, if the user privilege for *log on locally* is disabled, but *access this computer from the network* is enabled, the logon_method should be set to network.

Assigning Administrative Privileges

The configuration keyword, server_admin_id, has been introduced to allow the server administrator to assign administrative rights to other users in the server configuration file, edaserve.cfg.

Syntax: How to Assign Administrative Privileges

```
server_admin_id = user_ID[:password]{:SRV|:APP|:OPR}
```

where:

`server_admin_id`

Defines a comma-separated list of administrators. For each entry, the list includes the user ID, password, and administrative level. At least one member must be defined in the list. All users not in this list are limited to monitoring functions only.

`user_ID`

Is the ID of the user being granted administrative privileges.

`password`

Is an encrypted password, required if security mode is WCPROTECT or running on a Windows operating system for at least one member of the list.

`SRV`

Assigns Server Administrator privileges, giving the user the right to perform all administrative tasks available through the WebFOCUS Reporting Server Console, including the assignment of administrative or operator privileges to other users.

If more than one Server Administrator is defined, the credentials of the first valid member of the list is used for impersonating the FOCUS Database Server and other special services.

`APP`

Assigns Application Administrator privileges, giving the user the right to perform administrative tasks without the right to add, change, or remove many parameters.

`OPR`

Assigns Operator privileges, giving the user the right to perform limited administrative tasks, such as starting and stopping the server, and stopping agent, session, and connections.

Protecting the WebFOCUS Reporting Server Console

In addition to SECURITY ON and SECURITY OFF, the server supports a third security mode called WCPROTECT. This mode allows you to run the server without operating system security, while protecting access to the administrative functions of the WebFOCUS Reporting Server Console. In this mode, access to the WebFOCUS Reporting Server Console is protected by authenticating the connecting user against the Server Administrator user ID and password. This authentication is done without the involvement of any system services, but by comparison to encrypted configuration data.

To run the server in this mode, export the environment variable EDAEXTSEC = WCPROTECT before startup.

Windows Trusted Client

The WebFOCUS Reporting Server supports a Windows NT/2000 API client as a trusted client. This is optional and can take effect only on platforms where a password is not required. Since Windows requires a password to impersonate a user, this is not available on the Server for Windows. This option is controlled by a keyword in the server configuration file, edaserve.cfg. On z/OS and OS/390, this keyword is set in the service section of the server configuration file.

Syntax: How to Specify a Trusted Windows NT/2000 Client

```
trust_nt = {y|n}
```

where:

`y`

Supports Windows client as a trusted client.

`n`

Does not allow a trusted client. n is the default value.

HTTP Basic Authentication

In addition to the Integrated Windows Authentication (IWA) and server basic authentication methods, the HTTP listener supports HTTP basic authentication. This authentication method can be activated by setting the SECURITY keyword in the HTTP listener node block in the communication configuration file, odin.cfg.

```
NODE = LST_HTTP
BEGIN
 PROTOCOL = HTTP
 SERVICE = 8101
 CLASS = AGENT
 SECURITY = HTTPBASIC
END
```

This authentication method uses Web Server logon options. It is not recommended to use this method since it is not encrypted.

SSL/TLS Support

The HTTP Listener supports the following protocols through OpenSSL:

- Secure Sockets Layer (SSL) Version 3
- Transport Layer Security (TLS) Version 1

The listener supports DH and RSA algorithms for key exchange, DSS and RSA for authentication, 3DES and RC4 for encryption/decryption, MD5 and SHA1 for hashing.

To enable the SSL/TLS protocol, a certificate and a private key file must be specified in SSL_CERTIFICATE and SSL_PRIVATE_KEY in the communication configuration file, odin.cfg. The Listener presents the certificate to clients, and clients verify the certificate to determine if the listener can be trusted. Optionally, the listener can authenticate clients by providing trusted CA certificates in SSL_CA_CERTIFICATE. Only clients with certificates signed by one of the trusted CAs establishes a secure connection to the listener.

Lawson Security Adapter

In Release 5.2.3, you can apply Lawson security rules to the Lawson LAUA repository automatically. In previous releases, access was secured using DBA commands in the Master File or by controlling access with Managed Reporting. In Release 5.2.3, with the development of the Lawson Security Adapter, the appropriate Lawson security rules are automatically applied as defined in the Lawson LAUA repository. This is done by generating dynamic DBA commands and/or dynamic WHERE criteria.

The Lawson Security Adapter is configured using the WebFOCUS Reporting Server Console. The SUFFIX of the Lawson database (*xxxx*) and the default Lawson PRODUCTLINE (*yyyy*) need to be entered in the WebFOCUS Reporting Server Console. This information generates the code law_access = y in the server configuration file, edaserve.cfg, and the following three lines in the global server profile, edasprof.prf:

```
SET USER = LAWSON
ENGINE LAWSON SET LAW DBMS xxxx
ENGINE LAWSON SET LAW PRODLINE yyyy
```

Once the Lawson Security Adapter is configured, when the user makes a request against a Lawson table, the following queries and actions are performed:

- Is this a Lawson file?
- Is this a Lawson user?
- Does the user have access to this table?
- Generate DBA for any column restrictions.
- Generate WHERE criteria for any Lawson Company and Process Level restrictions.
- Generate WHERE criteria for any row restrictions.

For more information on configuring the Query Adapter for Lawson, see *Getting Started in Lawson* in the *iWay Data Adapter Administration for UNIX, Windows, OpenVMS, OS/400, OS/390 and z/OS* manual.

C2 Security Support

In Release 5.2.3, on Tru64 operating systems, server security implements the Security Integration Architecture (SIA), by default, because it supports both Enhanced (C2) and standard UNIX security. This C2 support extends only to Tru64.

The use of SIA blocks the server trusted connection feature even if Enhanced SIA (C2) is not active. To enable server trusted connections on Tru64 systems where C2 is not active, add the following keyword in the server configuration file, edaserve.cfg:

```
security_sia = n
```

Data Adapter Support

The following data adapters are fully supported on OS/390 and z/OS:

- ADABAS
- DATACOM
- DB2/CLI
- DB2/CAF (Call Attach Facility)
- Flat files
- FOCUS
- IDMS/SQL
- IDMS/DB
- IMS
- INFOMAN
- MILLENNIUM
- MODEL 204
- ORACLE
- TERADATA
- SAP
- SUPRA Server PDM (for PDM and RDM components)
- VSAM

Creating Synonyms

Create Synonym functionality is available for the following adapters on OS/390 and z/OS:

- ADABAS
- DB2/CLI
- DB2/CAF
- Fixed
- IMS
- TERADATA
- SAP
- VSAM

Synonyms are created through the Server Console and reside in a pre-specified directory on OS/390 and z/OS. Existing synonyms on z/OS and OS/390 can be accessed from OS/390 and z/OS but cannot be updated.

Accessing Files on z/OS and OS/390

How to:

Access Files Using the DYNAM Command

Example:

Accessing Files Using the DYNAM Command

In addition to accessing HFS-based files, Release 5.2 of the Server for OS/390 and z/OS can also access files on z/OS and OS/390 using the DYNAM command. You can use the DYNAM command to access partitioned data sets (PDSs) and data files on z/OS and OS/390 such as procedures (FOCEXECs), Master Files, and Access Files.

Syntax: How to Access Files Using the DYNAM Command

The DYNAM command has multiple variations in syntax, which are fully described in *FOCUS for Mainframe Overview and Operating Environments*.

The general syntax of the DYNAM command for allocation is

```
DYNAM ALLOCATE operand [operand]
```

where:

`ALLOCATE`

Allocates a data set. Can be abbreviated as ALLOC or ALLO.

`operand`

May be a keyword, a keyword followed by its parameter, or a parameter without a keyword.

Example: Accessing Files Using the DYNAM Command

The following examples use the DYNAM command with ALLOCATE to access different types of files:

```
DYNAM ALLOC FILE MASTER DA qualif.MASTER.DATA SHR, REUSE
DYNAM ALLOC FILE VSAM01 DA qualif.VSAM.CLUSTER SHR, REUSE
DYNAM ALLOC FILE CAR    DA qualif.CAR.FOCUS SHR, REUSE
```

External Sort

The z/OS and OS/390 external sort feature is fully supported on the Server for OS/390 and z/OS. External sort allows you to use dedicated sorting products such as SyncSort and DFSORT to generate large reports exceeding 5000 records much faster than if you were to use internal sort.

Syntax: How to Set the External Sort

To enable external sort, place the following SET command in the global server profile, edasprof.prf or in a procedure.

```
SET EXTSORT = {ON|OFF}
```

where:

`ON`

Enables external sort feature.

`OFF`

Turns external sort off. OFF is the default value.

Hiper Memory

Hiper Memory is a file management feature used to improve performance for temporary data sets on the server. Hiper Memory reduces server processing time by using hiperspaces to accelerate processing of temporary files. On the Server for OS/390 and z/OS, Hiper Memory settings are specified with a DYNAM command in the global server profile, edasprof.prf or in a procedure.

Syntax: How to Use DYNAM Commands With Hiper Memory

```
DYNAM MEMIO {ON|OFF|HIPER|MEMORY}
```

where:

`ON`

Enables Hiper Memory.

`OFF`

Disables Hiper Memory. I/O goes to disk. OFF is the default value.

`HIPER`

Uses HIPERSPACE memory for I/O.

`MEMORY`

Uses main storage for I/O.

Accessing z/OS and OS/390 Applications From the WebFOCUS Reporting Server Console

You can access applications residing on z/OS and OS/390 through the WebFOCUS Reporting Server Console for OS/390 and z/OS. You can view, execute, and directly edit applications from the console.

Available files, allocated in the EDASTART server JCL, include partitioned data sets (PDSs) and data files on z/OS and OS/390 such as procedures (FOCEXECs), Master Files, and Access Files. You can access these files through the Procedures page or the Metadata page in the WebFOCUS Reporting Server Console, reached by clicking the corresponding link on the left side of the start page.

Accessing z/OS and OS/390 VSAM Files Using XMI Server

How to:

Configure the XMI Listener and the XMI Client

Example:

Using the XMI Listener and XMI Client to Access Data on z/OS and OS/390

The Server for OS/390 and z/OS introduces read-only remote access to sequential data stored on z/OS and OS/390 using the XMI Server. This functionality consists of two parts:

- Multi-user XMI Listener running under server control.
- The client, running as an agent on any platform.

Procedure: How to Configure the XMI Listener and the XMI Client

To configure the XMI Listener and XMI Client to access data on z/OS and OS/390, complete the following steps.

1. For the XMI Listener, add the following node description in the communication configuration file, odin.cfg:

```
; XMI Database
Listener
NODE = XMISU
BEGIN
 PROTOCOL = TCP
 CLASS = SUBSERVER
 SERVICE = < # of Communication port >
END
```

2. For the XMI Listener, create the file xmisprof.prf to describe all VSAM files that clients are going to access. The description contains DYNAM statements where DDNAMEs are used by clients for access. For example,

```
DYNAM ALLOC DDNAME DBVSAM01 DSN DSNVSAM1 SHR REU
DYNAM ALLOC DDNAME DBVSAM02 DSN DSNVSAM2 SHR REU
 .
 .
 .
DYNAM ALLOC DDNAME DBVSAM0N DSN DSNVSAMN SHR REU
```

3. For the XMI Client, add the following node description in the communication configuration file, odin.cfg. The number of the communication port for the client and the XMI Listener must be the same.

```
; XMI Database Client
NODE = XMISU01
BEGIN
 PROTOCOL = TCP
 CLASS = SUCLIENT
 HOST = <Name of Host >
 SERVICE = < # of Communication port >
END
```

Example: Using the XMI Listener and XMI Client to Access Data on z/OS and OS/390

After the XMI Listener is running, the client can perform the following request:

```
USE DBVSAM02 on XMISU01
TABLE FILE DBVSAM02
 PRINT ......
END
```

Security Types

The Server for OS/390 and z/OS uses RACF, TOP SECRET, and ACF2 security packages. When configuring the server in secured mode, you can choose from two security settings, z/OS and OS/390 and INTERNAL. The z/OS and OS/390 setting prohibits the server from calling unauthorized external interfaces, while the INTERNAL setting allows calls to non-APF authorized interfaces.

Syntax: How to Specify a Security Type

To specify the security type, include the following setting in the Server JCL under the EDAENV DDNAME or in the profile on OS/390 and z/OS

```
SECURITY_TYPE = {INTERNAL|MVS}
```

where:

`INTERNAL`

Allows the server to call non-APF authorized external interfaces for IDMS. INTERNAL is the default value.

`MVS`

Prohibits the server from calling unauthorized external interfaces.

Server Accounting Using SMF Records

The Server for OS/390 and z/OS can write SMF accounting records that contain information on each connect and disconnect. You can view the accounting reports through the WebFOCUS Reporting Server Console. You must specify an SMF number to be used by the accounting facility when it sends records to the SMF system.

Syntax: How to Enable Accounting Using SMF Records

To activate usage accounting, add the following statement in the server configuration file, edaserve.cfg

```
smf_recno = n
```

where:

n

Indicates the SMF record number. This number represents the SMF number used by the accounting facility when it sends records to the SMF system. Values can range from 128 to 255; values less than 128 are not recommended.

Workload Manager Enclave Integration

On OS/390 and z/OS, a parameter in the service block of the server configuration file, edaserve.cfg, enables you to specify an enclave name. This enclave is a feature of Workload Manager (WLM).

Based on this setting, the task joins a Workload Manager enclave when a request starts and leaves it when the request finishes. This transfers control for the dispatching priority of the task to the Workload Manager. The rules for the enclave govern how the task performs with respect to running the request.

One use of this feature is to balance the workload so a long running request does not affect a short running request. This can be achieved through Workload Manager rules by lowering the priority of the long requests after a certain period of time. Without this feature, all requests share the regions priority.

To define the WLM service class that is to be used by all requests associated with a service, add the following keyword to the service block in the server configuration file, edaserve.cfg:

```
wlm_enclave_trname = WLM_service_class (up to 8 alphanumeric characters)
```

Support for Additional Operating Systems and Releases

Version 5 Release 2 adds support for the following operating systems and releases:

OS/400

- OS/400 V05R01 in addition to V04R04 and V04R05.

  Until fully certified, Release 5.2 should be considered candidate for release when used on OS/400 V05R02, IBM's latest OS level offering.

UNIX

- Compaq/ALPHA Tru64 V5.1
- SUN Solaris 8
- Linux 7.2 on Intel and OS/390 platforms

OpenVMS

- Compaq/ALPHA OpenVMS V7.3-1

Support for Large Files on HP and Solaris

On Solaris Version 5.6.2 or higher and HP Version 10.20 or higher, you can create and read non-FOCUS files with a size greater than two gigabytes, but less than 128 gigabytes, in either 32- or 64- bit mode.

Displaying Listener Statistics in the WebFOCUS Reporting Server Console

You can display listener statistics in the WebFOCUS Reporting Server Console. To do so, in the list of listeners, select the listener's Process ID.

MQ XML Listener

The MQ XML Listener listens on MQ for XML documents conforming to the iWay XML Document Specification. You can control the incoming XML from the listener for the purposes of execution, execution using RPC transform, or archiving.

Procedure: How to Configure the MQ XML Listener

The MQ XML Listener is configured using the WebFOCUS Reporting Server Console.

1. Open the Communications setup page, and in the Add New Node Block section specify the following:
   - Class of Agent.
   - Protocol of MQXML.
   - Desired name for the block.
2. Click *Continue*.
3. On the page that follows, supply qmanager, inqueue, outqueue, and errorqueue, click *Save and restart*.
4. Click the question mark for more information.

The server is now ready to accept XML documents from MQ.

SOAP XML Listener

The SOAP XML Listener listens on a TCP/IP port number for SOAP and XML documents conforming to the iWay SOAP and XML Document Specifications. You can now control the incoming XML from the listener for purposes of execution, execution through RPC transform, or archiving.

CHAPTER 8

Data Adapter Enhancements

This chapter describes all of the features for the Data Adapters in Version 5 Release 2 and has been updated to reflect Version 5 Release 2.3. Topics that contain Version 5 Release 2.3 features are marked with asterisks.

For complete documentation about data adapters, see the *Data Adapter Administration for UNIX, Windows, OpenVMS, OS/400, OS/390 and z/OS* or the *iWay Data Adapter Administration for MVS and VM* manual.

Topics:

Enhancements for Individual Adapters

* Data Adapter for Adabas

Data Adapter for Allbase on HP-UX

Data Adapter for Micro Focus C-ISAM

Data Adapter for DBFILE

* Data Adapter for DB2

Data Adapter for Enterprise Java Beans

* Data Adapter for Essbase

Data Adapter for Informix

Data Adapter for J.D. Edwards

* Data Adapter for Millennium

Data Adapter for MUMPS

* Data Adapter for PeopleSoft

Data Adapter for Progress

Data Adapter for SAP BW

* Data Adapter for SAP R/3

Data Adapter for Siebel

Data Adapter for UniVerse on HP-UX

* Data Adapter for VSAM

* Java Applications Adapter

Generic Adapter and Data Access Enhancements

DATASET Attribute for a Segment in a Master File

COBOL FD Translator Enhancements

Database Selection From the Console

* Field Identification as Read-Only

LIBPATH Display for DBMS

Long Qualified Field Names

Size Restrictions for Generated Alpha Fields

SQL Join Optimization

Text Fields for Relational Data Adapters

* SUBSTRING Function Behavior

VARCHAR Support

Data Adapter for Adabas

In this section:

Setting Connection Attributes

Adabas Transparent Superdescriptors

Adabas Hyper and Phonetic Descriptors

Adabas ORDER Fields in the Indexed Field List

Adabas Internal Sequence Numbers (ISN) and Global Format Buffer IDs (GFBID)

Switching Access Mode Using NOPACCESS

The Data Adapter for Adabas has been enhanced to support:

- Connection attribute settings through the WebFOCUS Reporting Server Console.
- Transparent Superdescriptors.
- Hyper and Phonetic Descriptors.
- ORDER fields.
- Internal Sequence Numbers (ISN).
- Global Format Buffer IDs (GFBID).
- Switching access modes with the SET NOPACCESS FIND command.

Setting Connection Attributes

Example:

Configuring the Data Adapter From the WebFOCUS Reporting Server Console Using Generic Parameters

Configuring the Data Adapter From the WebFOCUS Reporting Server Console Using OS/390 and z/OS Parameters

In Release 5.2.3, you can configure the Data Adapter for Adabas by setting connection attributes through the WebFOCUS Reporting Server Console. Configuring the adapter consists of specifying connection and authentication information for each of the connections you want to establish.

Example: Configuring the Data Adapter From the WebFOCUS Reporting Server Console Using Generic Parameters

The following Adabas parameters are applicable on all platforms and must be entered into the input fields of the Add Adapter screen:

1. Specify the Adabas database number to access. For example, *001*.
2. Specify the number of the Adabas Employee or another Demo File. For example, *011*.
3. Click *Configure* after the parameters are entered.

   Configuration results appears on the screen. The SET CONNECTION_ATTRIBUTES command is inserted into the edasprof.prf file. For example:

```
ENGINE ADBSINX SET CONNECTION_ATTRIBUTES ;1:ADAEMPL=11
```

Example: Configuring the Data Adapter From the WebFOCUS Reporting Server Console Using OS/390 and z/OS Parameters

The following Adabas parameters are applicable to OS/390 and z/OS platforms only, and must be entered into the input fields of the Add Adapter screen:

1. Specify the Adabas router SVC number. For example, *240*.
2. Specify the type of the device on which the Adabas Associator is located. For example, *3390*.
3. Specify the name of the Adabas source library that contains member ADALNK. For example, *ADABAS.V713.SRCE*.
4. Specify the name of the Adabas Associator Data Set. For example, *ADABAS.V713.EXAMPLE.DB001.ASSOR1*.
5. Click *Configure* after the parameters are entered.

   Configuration results appears on the screen. The SET CONNECTION_ATTRIBUTES command is inserted into the edasprof.prf file. For example:

```
ENGINE ADBSINX SET CONNECTION_ATTRIBUTES
;3:"ADAEMPL=1,ADASVC=240,ADADEVICE=3390,ADAASSO=ADABAS.V713.EXAMPLE.DB001.ASSOR1"
```

Adabas Transparent Superdescriptors

A superdescriptor is a key value associated with all or part of two to twenty Adabas fields. The Data Adapter for Adabas has been enhanced to transparently use the most efficient superdescriptor available, based on the nature of the request.

The SET SYNONYM command, used in conjunction with the CREATE SYNONYM command, generates the syntax required for transparent superdescriptors. The CREATE SYNONYM command is used to generate the metadata (Master File) for an Adabas file.

The SET SYNONYM command can be coded in any supported server profile.

Syntax: How to Set Synonyms for Superdescriptors

```
ENGINE ADBSINX SET SYNONYM {NEW|STD}
```

`NEW`

Describes superdescriptors as groups in the Master File. With NEW synonym format, neither the groups nor the components (fields) have field names specified in the Master File. Aliases are populated using the Adabas FDT (2-byte name).

When a synonym contains the NEW syntax, the Data Adapter for Adabas uses NEW logic to process the selection criteria of requests involving that synonym. Screening conditions within the request are analyzed and the superdescriptor that covers the highest order component fields required by the selection criteria are used.

Note that you must specify CALLTYPE = RL in the Access File in order to take advantage of superdescriptor-based access.

`STD`

Incorporates the usual CREATE SYNONYM behavior, which does not trigger the new logic for requests using superdescriptors. STD is the default value.

Adabas Hyper and Phonetic Descriptors

An Adabas file is defined with a set of indexes called descriptors. Descriptors are fields, partial fields, or groups of complete and partial fields used by Adabas to select records in a file. The Adabas term for these values, which reside in the File Associator Table, is inverted lists. Inverted lists contain the values of descriptors, a count of the total number of records in which each value appears, and the Internal Sequence Numbers (ISNs), in ascending order, associated with each occurrence.

The descriptors are identified to speed data location and to retrieve specific, frequently required data from Adabas files.

In Release 5.2.3, two additional types of descriptors are supported by the Data Adapter for Adabas:

- **Phonetic descriptors.** Consist of similar phonetic values for which an index has been created in Adabas. It is described at the field level in the Access File as TYPE = PDS. It can be used in selection tests, but it cannot be printed.

  The phonetic value of a descriptor is based on the first twenty bytes of the field value. Only alphabetic values are considered: numeric values, special characters, and blanks are ignored. Lowercase and uppercase alphanumeric characters are internally identical.

- **Hyper Descriptors.** Consist of values that are generated based on a user-supplied algorithm for which an index has been created in Adabas. It is described at the field level in the Access File as TYPE = HDS. It can be used in selection tests, but it cannot be printed.

  Any phonetic or hyper descriptor defined to the server in the Access File with TYPE = PDS or TYPE = HDS that has the following limitation: CALLTYPE = RL is not supported when this field is the IXFLD in an embedded cross-reference or JOIN.

  If you have selected a field defined with TYPE = NOP (for example, subdescriptors), TYPE = PDS, or TYPE = HDS, a FIND call is issued even if RL has been specified. For information on how to switch access modes from RL to FIND, see *Switching Access Mode Using NOPACCESS* on page 8-8.

Adabas ORDER Fields in the Indexed Field List

When the order of the data is significant, the ORDER field maintains the sequence number for each multiply occurring instance.

To use the ORDER attribute, include an additional field in your Master File with an ALIAS of ORDER. The ORDER field does not represent an existing Adabas field; it is used only for internal processing.

The SET ORDERKEYS command enables you to exclude ORDER fields from the list of indexed fields.

Syntax: How to Set ORDERKEYS in the Indexed Field List

```
ENGINE ADBSINX SET ORDERKEYS {ON|OFF}
```

where:

ON

Includes ORDER fields in the list of indexed fields. ON is the default value.

OFF

Excludes ORDER fields from the list of indexed fields.

Adabas Internal Sequence Numbers (ISN) and Global Format Buffer IDs (GFBID)

How to:

Use ISN and GFBID Parameters With CREATE SYNONYM

Reference:

Usage Notes For Adabas ISN Support

The Data Adapter for Adabas can employ a new data retrieval strategy using Read Logical by ISN (L1) calls to determine the Internal Sequence Number (ISN) of a record that was read or is inserted into an Adabas file.

ISN-based access is applicable only if an ISN field is described in the Master File. The name of the field is user-defined and has an ALIAS of ISN. This field can be defined only in a segment that contains non-repeating data (that is, using an access method defined in the Access File as ADBS):

```
FIELD = ISN_FIELD, ALIAS = ISN, I10, I4 ,$
```

The USAGE format is I and the ACTUAL format must be I4.

The ISN field can be defined manually in the Master File or using the CREATE SYNOYNM facility with the option PARMS ISN. For CREATE SYNONYM syntax, see *Use ISN and GFBID Parameters With CREATE SYNONYM* on page 8-8.

The Data Adapter for Adabas supports Global Format Buffer IDs (GFBID) to optimize the performance of queries that use the same Adabas field lists repeatedly. Field lists are generated in Adabas format buffers and can be retained.

GFBID support requires a GFBID field in the Master File. This field has a user-defined field name and an ALIAS of GFBID. It is used to determine the Global Format Buffer ID defined in read requests with identical field lists for the same database. The GFBID field has a USAGE format of A8 and an Actual format of A8. It can be defined only in segments that contain non-repeating data (that is, the access method defined in the Access File is ADBS).

- If the field list is changed but the same GFBID is used in a request, incorrect results may be displayed. In some cases a message about the possible mismatch between the Field Definition Table (FDT) and Master File will be issued.
- If two requests have the same list of selected fields but different fields are used in selection criteria, the requests must use different GFBID values.

The GFBID field can be defined manually in the Master File or using the CREATE SYNOYNM facility with the option PARMS GFBID. For CREATE SYNONYM syntax, see *Use ISN and GFBID Parameters With CREATE SYNONYM* on page 8-8.

If a GFBID field is defined in the Master File, it can be used in a request as part of the selection test. The adapter takes the GFBID value from the selection criteria, deactivate the field, and remove it from the match array. The GFBID value is placed in the ADD5 field of the Adabas Control Block for the request.

Remove these values from the Adabas queue by issuing the SET command

```
ENGINE ADBSINX SET GFBID_OFF ALL/<value> DBID <number>
```

`ALL`

Clears all GFBID values.

`<value>`

Clears a specific GFBID value.

When a single <value> is referenced in the SET command, all values issued for child segments are removed as well.

If <value> contains blanks, it must be enclosed in single quotation marks.

`<number>`

Is the database number.

Syntax: How to Use ISN and GFBID Parameters With CREATE SYNONYM

```
CREATE SYNONYM appname/synoynm
 FOR [FILE=] file-number[/predict-file-number[(predict-filename)]]
 DBMS ADBSINX
 DATABASE dbid[/predict-dbid]
 PARMS '<parameter_list>'
END
```

where:

parameter_list

Is the parameter value delimited by a space or a comma. Possible values are:

GFBID specifies that a field generated in the Master File is to be used to specify Global Format Buffer ID values.

ISN specifies that a field generated in the Master File is to be used to specify Internal Sequence Numbers (ISN).

NEW or STD is the processing mode for Adabas superdescriptor fields. (This is an alternative to the SET SYNONYM syntax described in *Set Synonyms for Superdescriptors* on page 8-4.) STD is the default value.

Reference: Usage Notes For Adabas ISN Support

Equality tests on the ISN field can be used to retrieve a single record when:

- The report request contains an equality operator in the selection test on an ISN list.
- The ISN field is used as the cross-referenced field in the Join to an Adabas file.

If the record defined in the test is not present in the Address Converter file, Adabas returns the Response Code 113. The adapter returns the following message, *Record is not found.*

It is also possible to assign a value for the ISN field when performing an Insert if the request contains the ISN field and the assigned value is not 0.

The adapter issues an Adabas N2 Direct Call to assign this ISN value to the inserted record. Adabas returns Response Code 113 if this value was already assigned to another record in the file or if it is larger than the MAXISN in effect for the file.

Switching Access Mode Using NOPACCESS

In Release 5.2.3, a customization parameter, SET NOPACCESS FIND, enables you to switch access mode from RL (Read Logical) to FIND in order to optimize performance when the NOP superdescriptor defined in a root segment is derived from fields that belong to different Master File segments. This option overrides the CALLTYPE = RL setting.

Syntax: How to Optimize Performance By Switching Access Mode

```
ENGINE ADBSINX SET NOPACCESS {MIXED|RL|FIND}
```

where:

MIXED

Allows the adapter to choose the access mode: RL or FIND. MIXED is the default value.

RL

Switches the access mode to Read Logical.

FIND

Switches the access mode from RL to FIND if a field defined with TYPE = NOP is selected (even if CALLTYPE = RL has been specified).

Data Adapter for Allbase on HP-UX

The Allbase data source is supported on the HP-UX platform. The Data Adapter for Allbase supports read/write access to Allbase data structures.

The Data Adapter for Allbase provides the following enhanced capabilities:

- Support for the CREATE SYNONYM command for the generation of Master Files.
- Direct SQL Passthru.
- SQL translation.
- Support for Allbase native data types.

Data Adapter for Micro Focus C-ISAM

The Micro Focus C-ISAM data source is supported on the RS/6000 platform. The Data Adapter for Micro Focus C-ISAM supports read access to Micro Focus C-ISAM data structures and the CREATE SYNONYM command for the generation of Master Files.

The Data Adapter for Micro Focus C-ISAM provides the following enhanced capabilities:

- Create Synonym (using COBOL Copy Book).
- SQL translation.
- Support for all Micro Focus C-ISAM compression types.

Data Adapter for DBFILE

The Data Adapter for DBFILE (SUFFIX = DBFILE) is used for physical, logical, and multi-format logical file support using the OS/400 OPNQRYF tool. These types of files are generally created through application languages such as RPG and Cobol. While considered legacy languages, they are very actively used today.

DBFILE Metadata

The Data Adapter for DBFILE has been enhanced to include adapter configuration and metadata creation from the WebFOCUS Reporting Server Console. You can also create metadata synonyms manually using the syntax:

```
CREATE SYNONYM abc FOR library/file DBMS DBFILE
END
```

While physical and logical files can also be accessed through the Data Adapter for DB2, performance tests show significant advantages— in the range of 300% (file size of 25021171 bytes, 130 fields, 37513 records)— when the Data Adapter for DBFILE is used.

Data Adapter for DB2

The Data Adapter for DB2 has been enhanced to support:

- Connection attributes settings from the WebFOCUS Reporting Server Console.
- FETCH FIRST *n* ROWS clause.
- Expanded DB2 table types on OS/400.
- DB2 Version 8.1 database.

Declaring DB2/CAF Connection Attributes From the WebFOCUS Reporting Server Console

In Release 5.2.3 on the USS server, the configuration screen for the Data Adapter for DB2/CAF displays the following fields:

Attribute	Description
SSID	DB2 SSID that is to be accessed, specified in the GENIDB2 job. This value must be uppercase.
Plan	Plan that was bound to DB2 using the GENIDB2 job. This value must be uppercase.
Execute DB2 BIND Command	Indicates whether to bind the program. Yes is the default value and adds the following fields to the configuration window: DSNCLST Library Name, Owner, Isolation Level.
DSNCLST Library Name	Is the name of the DSNCLST library. For example: DSN710.SDSNCLST. This attribute is available only if you choose to bind your program.
Owner	Determines the authorization ID of the Owner of the Plan. This attribute is available only if you choose to bind your program.
Isolation Level	Is the isolation level. CS is the default value. This attribute is available only if you choose to bind your program.

When you have completed the fields, click *Configure* to generate the adapter and build a DB2 plan. Click *Grant* to grant plan execution to public.

DB2 FETCH FIRST n ROWS

Example:

Reducing Answer Set Size

Starting with the following versions of DB2, READLIMIT tests restrict the size of the answer set returned:

- 5.0 UDB (UNIX, LINUX, Windows, and OS/2)
- 5.1 OS/400 (iSeries)
- 7.1 OS/390, z/OS

In prior releases, DB2 constructed a full answer set even if a request included a READLIMIT test. The adapter then retrieved from the answer set only the number of rows specified in the request.

When accessing a version of DB2 that supports the FETCH FIRST *n* ROWS clause, the data adapter translates a READLIMIT phrase to the appropriate clause to inform the RDBMS that only *n* rows need to be included in the answer set. For n = 1, the SQL request contains the clause FETCH FIRST 1 ROW ONLY. The reduction in answer set size reduces response time and enhances performance.

RECORDLIMIT tests do not restrict the size of the answer set returned by DB2. The full answer set is constructed and the data adapter retrieves the number of rows specified by the RECORDLIMIT test.

Note: This feature is not available in DB2 for VM or VSE.

Example: Reducing Answer Set Size

The following request turns on the STMTRACE trace component and issues a retrieval request with a READLIMIT phrase. The SQL request contains the FETCH FIRST 2 ROWS clause:

```
SET TRACEUSER = ON
SET TRACEOFF = ALL
SET TRACEON = STMTRACE//CLIENT
TABLE FILE EMPINFO
 PRINT LAST_NAME FIRST_NAME CURRENT_SALARY
 BY EMP_ID
 IF READLIMIT EQ 2
END
```

The generated SQL request contains the FETCH FIRST 2 ROWS clause:

```
SELECT T1."EID",T1."LN",T1."FN",T1."CSAL" FROM
USER1."EMPINFO" T1 ORDER BY T1."EID" FETCH FIRST 2 ROWS ONLY;
If the test specifies READLIMIT EQ 1, the SQL request contains the clause
FETCH FIRST 1 ROW ONLY:
SELECT T1."EID",T1."LN",T1."FN",T1."CSAL" FROM
USER1INFO" T1 ORDER BY T1."EID" FETCH FIRST 2 ROW ONLY;
```

DB2 Table Types on OS/400

The WebFOCUS Reporting Server Console CREATE SYNONYM facility for DB2 on OS/400 has been extended beyond tables and views to include types for physical and logical files, which can also be accessed by DB2.

Support for DB2 8.1

Release 5.2.3 supports the DB2 Version 8.1 database on UNIX and NT.

Data Adapter for Enterprise Java Beans

The Data Adapter for Enterprise Java Beans supports Create Synonym and Access Beans.

CREATE SYNONYM uses the Interrogate Method to retrieve bean property, method, or event information. If the bean returns data, the Master File contains a field description.

A table request uses the Invoke Method to access the bean. The value of the bean Primary Key is specified in a WHERE clause. Only one active Primary Key is allowed in the request. The URL of the server where the bean is located is stored in the Access File.

The adapter currently supports access to Enterprise Java Beans deployed under the BEA WebLogic and IBM WebSphere Application servers.

Data Adapter for Essbase

In this section:

Describing the Measures Dimension in the Master File

Summing on Non-Aggregated Fields

Preventing Aggregation of Non-Consolidating Members

Suppressing Shared Members

Suppressing Zero Values

Suppressing Missing Data

Suppressing Zero Values and Missing Data

Support for User-Defined Attributes

The Data Adapter for Essbase has been enhanced with the following new features:

- The CREATE SYNONYM command supports parent/child view.
- The SET MEASURE command allows the server to recognize any one dimension as an accounts/measures dimension without having to change the ESSBASE outline.
- The SET RESTRICTSUM command allows SUM operations on non-aggregated fields that appear in the Access File.
- The SET AGGREGATE NOOP command controls the use of SUM operations on Essbase members with the (~) consolidation property.
- The SET SUPSHARE command allows Shared Members to be suppressed.
- The SET SUPZEROS command permits zero values to be suppressed during reporting.
- The SET SUPMISSING command permits empty cells (missing data) to be suppressed during reporting.
- The SET SUPEMPTY command permits an entire row that contains either zero values or missing data to be suppressed during reporting.
- The ESSBASE OLAP Server's User Defined Attributes (UDAs) can be used in requests.

Describing the Measures Dimension in the Master File

The SET MEASURE command enables you to set or change the Accounts tagged dimension without having to change the outline in the Essbase Database Server. With this setting, the server produces an actual field name for every member of the named dimension in the Master File rather than producing the dimension in terms of generations. The SET MEASURE command must be executed before the synonym is created.

Syntax: How to Set or Change the Measures Dimension in the Master File

To set the Accounts tag for a dimension, use the syntax

```
ENGINE ESSBASE SET MEASURE dimension_name FOR synonym
```

where:

dimension_name

Is the name of the dimension to be interpreted as an Accounts tagged dimension when generating a synonym.

synonym

Is an alias for one *application.database* combination.

Summing on Non-Aggregated Fields

Any two-pass calculated members that are found in the accounts (or Scenario) dimension and members with the (-), (\), (*), and (%) consolidation properties in the Essbase outline, have the following attributes in the associated Access File:

```
MEMBER = membername, AGGREGATE = NO ,$
```

Due to the nature of two-pass calculation members, a summation request (SQL SUM... or SUM in TABLE) may produce incorrect values. Therefore, if the member has the AGGREGATE = NO attribute in the Access File, a SUM action in the request against the member results in the following message:

```
(FOC43241) Aggregation is requested for non aggregatable member(s)
```

The server allows SUM to be used on non-aggregated fields in the Access File.

Syntax: How to Turn Aggregation ON/OFF For Non-Aggregated Fields

```
SQL ESSBASE SET RESTRICTSUM {ON|OFF} FOR synonym
```

where:

ON

Does not allow the use of SUM on non-aggregated fields. ON is the default value.

OFF

Allows the use of SUM on non-aggregated fields.

synonym

Is an alias for the data source and is a maximum of 64 characters.

Preventing Aggregation of Non-Consolidating Members

In Release 5.2.3, member consolidation properties determine how children roll up into the parents in the Essbase outline. The members with the (~) as a consolidation property in the Essbase outline, are *not* rolled up in the database. The SET AGGREGATE NOOP (No Operation) setting only affects those Essbase members in the Accounts dimension with the (~) consolidation property. If the AGGREGATE NOOP setting is OFF, SUM is not permitted on Essbase members with the (~) consolidation property and those members appear in the Access File as:

```
MEMBER = ADDITIONS,  AGGREGATE = NO ,$
```

Syntax: How to Prevent Aggregation of Non-Consolidating Members

You must set AGGREGATE NOOP to OFF before you create the synonym

```
SQL ESSBASE SET AGGREGATE NOOP {ON|OFF}
```

where:

ON

Allows the use of SUM on NOOP Essbase members. With this setting, Essbase members are not added to the Access File. ON is the default value.

OFF

Does not allow the use of SUM on NOOP Essbase members. Members with the (~) consolidation property are added to the Access File.

Note: Any member of the Accounts dimension in the Essbase outline tagged with (-), (/), (*), or (%) as a consolidation property automatically appears in the Access File with AGGREGATE = NO.

Suppressing Shared Members

Shared members in Essbase store pointers to data that is stored in the real member. Therefore, although the data is shared between the two members, it is only stored one time. You can exclude shared members from reports using the SET SUPSHARE command.

Syntax: How to Suppress Shared Members in Essbase

```
SQL ESSBASE SET SUPSHARE {ON|OFF}
```

where:

ON
: Excludes shared members in a report.

OFF
: Includes shared members in a report. OFF is the default value.

Suppressing Zero Values

In Release 5.2.3, Essbase stores zero values that you can exclude from reports using the SET SUPERZEROS command. The setting applies to an entire row. If one member in a row has a value of zero, and the other values in the same row have values other than zero, the zero value is not suppressed.

Syntax: How to Suppress Rows With Zero Values

```
SQL ESSBASE SET SUPZEROS {ON|OFF}
```

where:

ON
: Suppresses an entire row that contains 0 values.

OFF
: Does not suppress zero values. OFF is the default value.

Suppressing Missing Data

In Essbase, empty cells are known as missing or #MISSING data. In Release 5.2.3, you can suppress missing data during reporting using the SET SUPMISSING command.

Syntax: How to Suppress Rows With Missing Data

```
SQL ESSBASE SET SUPMISSING {ON|OFF}
```

where:

ON
: Suppresses an entire row that contains #MISSING data.

OFF
: Does not suppress rows with #MISSING data. OFF is the default value.

Suppressing Zero Values and Missing Data

In Release 5.2.3, you can use the SET SUPEMPTY command to suppress an entire row that contains either zero values, missing (#MISSING) data, or a combination of the two.

Syntax: How to Suppress Rows With Zero Values or Missing Data

```
SQL ESSBASE SET SUPEMPTY {ON|OFF}
```

where:

ON

Suppresses an entire row that contains either zero values, #MISSING data, or a combination of the two.

OFF

Does not suppress rows that contain either zero values or #MISSING data. OFF is the default value.

Support for User-Defined Attributes

A User-Defined Attribute (UDA) in Essbase enables you to select and report on data based on a common characteristic. You can include User-Defined Attributes in report requests. For each dimension with UDAs in an Essbase outline, CREATE SYNONYM generates a UDA field name under a segment called UDA in the Master File.

Example: Reporting From a UDA Segment

The Sample Master File contains the following UDA segment:

```
$ DIMENSION: UDA
SEGMENT = UDA, SEGTYPE = U, PARENT = BASIC ,$
FIELD = MARKET_UDA,
ALIAS = Market,
USAGE = A13,ACTUAL = A13 ,$
```

When referencing UDAs in a request, you must also reference a member of the dimension that contains the UDA. In this example, STATE is a member of the MARKET Dimension in the Master File.

```
TABLE FILE SAMPLE
 PRINT SALES BY STATE
 WHERE MARKET_UDA EQ 'New Market'
END
```

Data Adapter for Informix

The Data Adapter for Informix supports the LVARCHAR data type. On the server, an LVARCHAR column is mapped as ALPHA.

Data Adapter for J.D. Edwards

In Release 5.2.3, the Data Adapter for J.D. Edwards allows WebFOCUS and other applications to access J.D. Edwards data sources. The adapter converts data or application requests into native J.D. Edwards statements and returns optimized answer sets to the requesting program.

Data Adapter for Millennium

In Release 5.2.3, the Data Adapter for Millennium running on the z/OS and OS/390 Server supports compressed Millennium VSAM files. The Millennium decompression routines have been added to the Data Adapter for VSAM through the existing ZCOMP1 exit point.

To enable support for compressed files, include the following SET command in your server profile, EDASPROF.prf:

```
ENGINE CPMILL SET ZCOMP FOCMILZ
```

The adapter requires two control files to be built and allocated to DDNAME CPMILL and CPMILLI using JCL or DYNAM allocation. The Millennium control file is used as input to these files and is dynamically allocated to the server only for the creation of CPMILL and CPMILLI.

For details on how to create the required control files, see the *iWay Server Configuration and Operations for MVS, Version 5.2* manual.

Data Adapter for MUMPS

The Data Adapter for MUMPS is available for OpenVMS 7.2-2 and 7.3-1 running DSM Mumps 7.3. The adapter has been enhanced to support:

- Read and write capabilities.
- Creation of server metadata (synonym).
- Transaction processing with recovery units. This allows you to protect the logical consistency of your application database during a failure.
- Extended descriptions of data values for any level of subscripts.
- Field delimiters.
- Fixed and variable field redefinitions.
- Join between logical statements.

Data Adapter for PeopleSoft

In Release 5.2.3, the Data Adapter for PeopleSoft provides data access for reporting from the following PeopleSoft record types: SQL tables, SQL views, and query views. The adapter fully enforces PeopleSoft Query tree and low-level security.

The adapter supports both the client-server architecture of PeopleSoft 7.5 and earlier, as well as the newer PeopleSoft Pure Internet Architecture.

The adapter must be configured on the server used for PeopleSoft reporting *prior* to creating any connections to PeopleSoft data sources.

The adapter provides the following three classes of services:

- Authentication services.
- PeopleSoft data source connection management services.
- Metadata administration and data access security services.

Native data source SQL scripts are provided to prepare the following PeopleSoft data constructs for simplified reporting:

- Effective dated records.
- PeopleSoft trees.

Data Adapter for Progress

The Data Adapter for Progress has been enhanced to use an ODBC connection to the database. ODBC provides full support for ANSI SQL-92 specifications, which supports Binary Large Objects (BLOBs). The Data Adapter for Progress 9.1d is supported on the following platforms:

- Windows NT and Windows 2000
- IBM AIX
- Sun Solaris
- HP-UX
- Linux

The adapter is available on Compaq tru64 platforms only as an embedded SQL-89 interface.

Data Adapter for SAP BW

The Data Adapter for SAP BW is a data adapter from SAP with consolidated information from SAP R/3 and external sources. BW is pre-configured for, but not limited to, SAP R/3 data and is based on the same business objects and metadata as R/3.

BW reporting is based on the OLE DB for OLAP model. OLE DB is a successor to ODBC (Microsoft's Open Database Connectivity standard) designed to access multi-dimensional datastores though Multi-Dimensional Expressions (MDX).

The OLAP model defines key figures (numerics) and characteristics (alpha) to be used in multiple combinations for data analysis.

The Data Adapter for SAP BW offers:

- Retrieval of all BW multi-dimensional metadata objects.
- SAP variable support for parameterized queries.
- SAP attribute (property) support for complete reporting.
- Support for SAP key and name fields.
- Reporting from InfoCubes and ODS objects.
- Built-in flattening for rapid conversion of data sets to row sets.
- High speed performance.
- Automated WebFOCUS Reporting Server Console for server metadata creation.

Data Adapter for SAP R/3

Enhancements to the SAP R/3 Native Interface include:

- Security.
- Business Application Programming Interface (BAPI) support.
- Join support.
- Aggregation support.

Enhancements to IDoc Processing include:

- IDoc Listener.

Security Support for the SAP R/3 Native Interface

The Native Interface for SAP R/3 has been enhanced to handle the user/password provided in the following statement:

```
SQL SQLSAP SET CONNECTION_ATTRIBUTES system/user,password: 'client'
```

This feature allows you to run a secured request that involves the following:

- One or more BAPIs, as BAPIs are secured by SAP.
- SAP delivered logical databases, as the server code is secured by SAP.
- Function Modules (either SAP or customer) that are fully prototyped and that include the proper AUTHORITY-CHECK statements.

Reference: Usage Notes for Security in SAP R/3 Native Interface

If the user ID and password are not provided in the connection string, the request is executed with the user ID and password provided in the sapserv.cfg file for the given system.

If user ID and password are provided in the connection string, all work except execution of the request is done with the user ID and password stored in the sapserv.cfg file. The user ID and password from the connection string are used only on execution of the request.

BAPI Support in the SAP R/3 Interface

The current release supports most read-only BAPIs, including joins, as described in the following example:

```
CREATE SYNONYM baseapp/BUS0002_GETLIST
 FOR BUS0002/GETLIST
 BAPI DBMS SQLSAP AT I46
END
CREATE SYNONYM baseapp/BUS0002_GETDETAIL
 FOR BUS0002/GETDETAIL BAPI
 DBMS SQLSAP AT I46
END
JOIN BAPI0002_COMP_CODE IN COMPANYCODE_GETLIST TO
CCGD2_COMP_CODE IN COMPANYCODE_GETDETAIL
END
TABLE FILE COMPANYCODE_GETLIST
 PRINT
 CCGL0_TYPE NOPRINT
 BAPI0002_COMP_CODE
 BAPI0002_COMP_NAME
 CCGD2_CURRENCY
 CCGD2_LANGU
 IF BAPI0002_COMP_CODE NE '2300' OR '6000'
END
```

Join Support in the SAP R/3 Interface

The Native Interface supports all from-SAP/to-SAP joins. However, for performance reasons, it is not recommended to join to SAP from a non-SAP data source.

It is more efficient to hold the keys in a sequential file, and then use the following code:

```
TABLE FILE SAP
 PRINT FIELDS
 IF KEYS IS (HOLD)
END
```

Aggregation Support in the SAP R/3 Interface

In Release 5.2.3, the Native Interface for SAP R/3 has been enhanced to pass aggregation to Open/SQL for Transparent Tables. To activate this functionality, use the following SET command:

```
SQL SQLSAP SET OPTIMIZATION SQL
```

IDoc Listener

Intermediate Documents (IDocs) and the IDoc Listener provide a means to perform Electronic Data Interchange (EDI) with SAP. EDI-enabled applications in SAP are capable of generating IDoc data from an SAP document. IDocs can be sent either in batch or real time, and are perfect for online data warehousing.

The purpose of the IDoc Listener is to generate a Relational Data Base Management System (RDBMS) environment that replicates the source SAP RDBMS tables from which the IDoc originates. Those tables are then updated automatically as new IDocs are received.

For more information on configuration and operations, see the *IDoc Listener* documentation. To access this document, open your WebFOCUS Reporting Server Console, follow the Summary of New Features link on the main page, then follow the link provided. To access this document, open your WebFOCUS Reporting Server Console, follow the Summary of New Features link on the main page, then follow the link provided.

Data Adapter for Siebel

The Data Adapter for Siebel is available for UNIX, NT, and z/OS. This read-only adapter supports the following capabilities:

- Server metadata creation from Business Component and Business Object.
- DML and SQL syntax.
- Multiple connections.
- Joins.
- Visibility mode.
- Siebel data types, including integer, date, time, datetime, and alpha. All others are mapped to alpha.
- Multi-valued groups.

Data Adapter for UniVerse on HP-UX

The UniVerse data source is supported on an HP-UX platform. The Data Adapter for UniVerse supports read/write access to UniVerse data structures and supports the CREATE SYNONYM command for the generation of Master Files. The Data Adapter for UniVerse provides the following enhanced capabilities:

- Direct SQL Passthru.
- Support for the CREATE SYNONYM command.
- SQL Translation.
- Support for UniVerse native data types.

Data Adapter for VSAM

In Release 5.2.3, the most efficient way to retrieve selected records from a VSAM KSDS data source is by applying an IF screening test against the primary key. This results in a direct read of the data using the data source's index. Only those records that you request are retrieved from the file. The alternative method of retrieval, the sequential read, forces the data adapter to retrieve all the records into storage.

Selection criteria that are based on the entire primary key, or on a subset of the primary key, cause direct reads using the index. A partial key is any contiguous part of the primary key beginning with the first byte.

The EQ and IS relations realize the greatest performance improvement over sequential reads. When testing on a partial key, equality logic is used to retrieve only the first segment instance of the screening value. To retrieve subsequent instances, NEXT logic is used.

Screening relations GE, FROM, FROM-TO, GT, EXCEEDS, IS-MORE-THAN, and NOT-FROM-TO all obtain some benefit from direct reads. The following example uses the index to find the record containing primary key value 66:

```
IF keyfield GE 66
```

It then continues to retrieve records by sequential processing, because VSAM stores records in ascending key sequence. The direct read is not attempted when the IF screening conditions NE, IS-NOT, CONTAINS, OMITS, LT, IS-LESS-THAN, LE, and NOT-FROM are used in the report request.

Similar performance improvement is available for ESDS and KSDS files that use alternate indexes. An alternate index provides access to records in a key-sequenced data set based on a key other than the primary one. All benefits and limitations inherent with screening on the primary or partial key are applicable to screening on the alternate index or partial alternate index.

Java Applications Adapter

In this section:

Writing a Java Class

Java Class Communications

Compiling and Running a Java Program

How to:

Use CALLJAVA to Execute a Java Class

Use EX to Execute a Java Class

In Release 5.2.3, you can easily access a Java class in your application, much as you would access a program with CALLPGM. There are two ways to call a Java class:

- CALLJAVA call.
- EX command.

Either method enables you to pass parameters to the Java class.

Syntax: How to Use CALLJAVA to Execute a Java Class

```
CALLJAVA class, parameter1, parameter2, ...
```

where:

class

Is the full name of the class to be invoked.

parameter1, parameter2, ...

Are the remaining parameters which must be passed to the Java class.

Syntax: How to Use EX to Execute a Java Class

```
EX java.class parameter1, parameter2, ...
```

where:

java.class

Is the full name of the class to be invoked and must be preceded by the prefix java.

parameter1, parameter2, ...

Are the parameters which must be passed to the Java class.

Writing a Java Class

When you write a Java class to be invoked by the server, you use the class with the CALLJAVA interface, much as you would use a 3GL program with the CALLPGM interface. The CALLJAVA interface defines two methods, execute and fetch:

- **Execute method.** Receives three parameters: user ID, password ID, and the String array of parameters. Any one of those parameters can be a null object reference. Null reference for the parameters array represents invocation with no parameters. The server invokes the Java class in the "password pass through" mode. The execute method performs the request and returns the instantiated IBI Answer Set object populated with the answer set description. The fetch method populates this object with data.
- **Fetch method.** Invoked by the server to receive one row of the answer set at a time. IBI_EOD is returned when answer is finished; IBI_DATA is returned to indicate more data is coming.

You can use ibtrace.println to trace execution of the program and ibtrace.printException to output all information about a caught interruption. The output of ibtrace class method calls is forwarded into the standard server trace file.

Java Class Communications

When executing a Java class (either with the CALLJAVA or EX command), the server and the program communicate using an IBI answer set object. This object has to be instantiated and populated with the answer set description on an "execute" method call. The method is called by the server only once. The server calls a "fetch" method repeatedly until it receives an IBI_EOD indicator. The server expects to receive the answer set row by row in the same instance of the IBI answer set object.

Compiling and Running a Java Program

When compiling your Java program, the jscom3.jar file located in the etc directory in the EDAHOME directory needs to be accessible using the CLASSPATH environment variable or the javac command parameter.

When executing your Java class, place the client jar file containing the Java class to be invoked in the CLASSPATH environment variable prior to starting the server.

DATASET Attribute for a Segment in a Master File

In this section:

DATASET Behavior in FOCUS Data Sources

How to:

Use the DATASET Attribute

Example:

Allocating a FOCUS Segment Using the DATASET Attribute

In addition to having a DATASET attribute at the file level in a FOCUS Master File, you can add the DATASET attribute to a segment declaration to specify the physical file name for a LOCATION segment or a cross-referenced segment with field redefinitions. In addition, the DATASET attribute permits you to bypass the search mechanism for default data source location. DATASET eliminates the need to allocate data sources using JCL, FILEDEF, DYNAM, and USE commands.

User allocation and system specific behavior is as follows:

Platform	User allocation command
CMS/UNIX/Windows NT	FILEDEF
TSO	DYNAM ALLOC or TSO ALLOC

DATASET Behavior in FOCUS Data Sources

The DATASET attribute can be used on the file or segment level of a FOCUS Master File. If a USE command or explicit allocation is issued for the file, a warning is issued that the DATASET attribute will be ignored.

The segment with the DATASET attribute must be either a LOCATION segment or a cross-referenced segment. For cross-referenced segments:

- If field declarations are specified for the cross-referenced fields, the DATASET attribute is the only method for specifying a physical file because FOCUS does not read the cross-referenced Master File and, therefore, does not pick up its DATASET attribute if one is specified.
- If field declarations are not specified for the cross-referenced fields, it is better to place the DATASET attribute at the file level in the cross-referenced Master File as FOCUS reads and parses the cross-referenced Master File. In this case, specifying different DATASET values at the segment level in the host Master File and the file level of the cross-referenced Master File causes a conflict, resulting in a (FOC1998) message.

If DATASET is used in a Master File whose data source is managed by the FOCUS Database Server, the DATASET attribute is ignored on the server side because the FOCUS Database Server does not read Master Files for servicing table requests.

The DATASET attribute in the Master File has the lowest priority:

- A user's explicit allocation overrides DATASET attributes.
- The USE command for FOCUS data sources overrides DATASET attributes and explicit allocations.

Note: If a DATASET allocation is in effect, a CHECK FILE command must be issued in order to override it by an explicit allocation command. The CHECK FILE command deallocates the allocation created by DATASET.

Syntax: How to Use the DATASET Attribute

For a LOCATION segment:

```
SEGNAME = segname, SEGTYPE = segtype, PARENT = parent,
 LOCATION = filename, DATASET = 'physical_filename [ON sinkname]' ,$
```

For a cross-referenced segment

```
SEGNAME = segname, SEGTYPE = segtype, PARENT = parent,
 [CRSEGNAME = crsegname,] [CRKEY = crkey,] CRFILE = crfile,
 DATASET = 'filename1 [ON sinkname]', FIELD = ...
```

where:

filename

Is the logical name of the LOCATION file.

physical_filename

Is the platform-dependent physical name of the data source.

sinkname

Indicates that the data source is located on the FOCUS Database Server. This attribute is valid for FOCUS data sources.

On z/OS and OS/390, the syntax is:

```
{DATASET|DATA} = 'qualifier.qualifier ...'
```

or

```
{DATASET|DATA} = 'ddname ON sinkname'
```

On CMS, the syntax is:

```
{DATASET|DATA} = 'filename filetype filemode [ON sinkname]'
```

On UNIX, the syntax is:

```
{DATASET|DATA} = 'path/filename'
```

On Windows, the syntax is:

```
{DATASET|DATA} = 'path\filename'
```

Example: Allocating a FOCUS Segment Using the DATASET Attribute

On z/OS and OS/390:

```
FILE = ...
SEGNAME = BODY,SEGTYPE = S1,PARENT = CARREC,LOCATION = BODYSEG,
DATASET = 'USER1.BODYSEG.FOCUS',
FIELDNAME = BODYTYPE,TYPE,A12,$
FIELDNAME = SEATS,SEAT,I3,$
FIELDNAME = DEALER_COST,DCOST,D7,$
FIELDNAME = RETAIL_COST,RCOST,D7,$
FIELDNAME = SALES,UNITS,I6,$
```

On UNIX, OS/390, and z/OS:

```
FILE = ...
SEGNAME = BDSEG,SEGTYPE = KU,CRSEGNAME = IDSEG,CRKEY = PRODMGR,
CRFILE = PERSFILE,DATASET ='/u2/prod/user1/idseg.foc',
FIELD = NAME,ALIAS = FNAME,FORMAT = A12,INDEX = I ,$
```

On z/OS and OS/390 with SU:

```
FILE = ...
SEGNAME = BODY,SEGTYPE = S1,PARENT = CARREC,LOCATION = BODYSEG,
DATASET = 'BODYSEG ON MYSU',
FIELDNAME = BODYTYPE,TYPE,A12,$
FIELDNAME = SEATS,SEAT,I3,$
FIELDNAME = DEALER_COST,DCOST,D7,$
FIELDNAME = RETAIL_COST,RCOST,D7,$
FIELDNAME = SALES,UNITS,I6,$
```

COBOL FD Translator Enhancements

You can create server metadata descriptions by translating COBOL FD definitions for data adapters, including VSAM, C-ISAM, and flat files. New functionality allows you to:

- Select COBOL Copy Book by name.
- Select COBOL Copy Book from a list by providing the path to the directory.
- Specify the following customization options for metadata generation from the Web Console:

Option	Description
On Error	Enter *Continue* to continue generating the Master File when an error occurs. Enter *Abort* to stop generating the Master File when an error occurs. Enter *Comment* to produce a commented Master File when an error occurs. Continue is the default value.
Hyphens as	Enter *No* to remove all hyphens in the COBOL name from the Master File field names. Enter *Yes* to replace all hyphens in the COBOL name with the underscore character. Yes is the default value.
Redefines	You may treat COBOL REDEFINE fields in one of the following ways: • Choose *Segments* to describe REDEFINE fields as segments in the Master File. Segments is the default value. • Choose *Comments* to describe REDEFINE fields as comments in the Master File. • Choose *None* to exclude REDEFINE fields altogether.
Occurs as	Choose *Segments* to describe OCCURS structures as segments. Otherwise, choose *Field*. Segments is the default value.
Alignment	Enter *Yes* to insert slack bytes into records to ensure the alignment of numeric fields. Enter *No* to generate Master Files without alignment of slack bytes. Yes is the default value.
Number of Hyphens to skip	The Translator removes characters from the left, up to and including the *n*th hyphen, where the value of *n* can be 1 or 2. *0* retains the entire COBOL name. *All* means all prefixes are removed. 0 is the default value.

Option	Description
Order Fields	Enter *Yes* to generate Order fields in a Master File. Enter *No* to generate a Master File without Order fields. No is the default value.
Level 88 as	Enter *Comment* to include COBOL Level 88 fields as comments in the Master Files. Enter *Skip* to exclude level 88 fields. Skip is the default value.
Zoned Numeric Fields	Choose *Packed* to describe zoned numeric fields as packed numeric. Choose *Alpha* to describe zoned numeric fields as alphanumeric. Alpha is the default value.
Numeric Field Edit Options	
Zeros: Enter *Suppress* to suppress printing of the digit zero for a field whose value is zero. Enter *Display* to display leading zeros, for example, 00124. Enter *None* for no formatting. None is the default value.	
Negative value: Enter *Bracket* to bracket negative values, for example, (1234). Enter *Credit* to credit negative values, for example, 1234 CR. Enter *None* for no formatting. None is the default value.	
Dollar Sign: Enter *Floating* to display a floating dollar sign and commas, for example, $1,234. Enter *Fixed* to display a fixed dollar sign and commas, for example, $ 1,234. Enter *None* for no formatting. None is the default value.	
Separate Thousands: Enter *Comma* to include commas where appropriate. Enter *None* for no formatting. None is the default value.	

Database Selection From the Console

In Release 5.2.3, for Microsoft SQL Server or Sybase Adaptive Server Enterprise (ASE), you can choose the database to work with directly from the WebFOCUS Reporting Server Console using the *Select Database* option.

Field Identification as Read-Only

How to:

Specify a Field as Read-Only in a Master File

Example:

Using FIELDTYPE = R

The field-level Master File attribute FIELDTYPE = R identifies a field as read-only. This attribute provides support for relational data sources with auto-increment columns such as IDENTIFY or timestamp columns in DB2. These columns are automatically incremented by the RDBMS.

CREATE SYNONYM adds the FIELDTYPE = R attribute for read-only fields in an RDBMS table.

The update languages MODIFY and Maintain do not populate columns with this designation. If an update command is issued for a read-only field, it is ignored by MODIFY and Maintain.

Note: This attribute is not supported for DB2 on VM.

Syntax: How to Specify a Field as Read-Only in a Master File

```
FIELD = ..., FIELDTYPE = R ,$
FIELD = ..., FIELDTYPE = RI ,$
FIELD = ..., FIELDTYPE = IR ,$
```

where:

`R`

Indicates that the field is read-only. Any number of fields can have this attribute.

`RI,IR`

Indicates that the field is read-only and indexed (applies to FOCUS, VSAM, and XFOCUS data sources only).

Example: Using FIELDTYPE = R

Autoinc1 is an Microsoft SQL Server table with an IDENTITY field called Control as the first field, which is a key.

The following CREATE SYNONYM command creates a Master File with the FIELDTYPE = R attribute for the Control field:

```
CREATE SYNONYM MNTAUTO FOR AUTOINC1 DBMS SQLMSS AT MSXYZ
END
```

The generated Master File is:

```
FILE = MNTAUTO , SUFFIX = SQLMSS ,$
SEGNAME = MNTAUTO , SEGTYPE = S0 ,$
FIELD = CONTROL   ,CONTROL   ,I11 ,I4  ,MISSING = OFF, FIELDTYPE = R ,$
FIELD = LASTNAME  ,LASTNAME  ,A12 ,A12 ,MISSING = OFF ,$
FIELD = FIRSTNAME ,FIRSTNAME ,A12 ,A12 ,MISSING = OFF ,$
FIELD = ITEM      ,ITEM      ,A20 ,A20 ,MISSING = OFF ,$
FIELD = AMOUNT    ,AMOUNT    ,P19 ,P10 ,MISSING = OFF ,$
```

The generated Access File is:

```
SEGNAME = MNTAUTO ,
TABLENAME = AUTOINC1,
CONNECTION = MSSXYZ,
KEYS = 1
,$
```

LIBPATH Display for DBMS

When configuring a DBMS adapter on UNIX environments, the Shared Library Path is displayed on the Adapter ADD panel.

Long Qualified Field Names

Example:

Using Long Qualified Field Names

A qualified field consists of a file name, a segment name, and a field name, separated by qualifying characters (QUALCHARs). A long qualified field name is comprised of a long file name, a long segment name, and a long field name. Long qualified field names are useful for storing documents that use large descriptive field names such as those used and transmitted with XML.

For a relational or an XFOCUS data source, file names can be up to 64 characters, segment names up to 64 characters, and field names up to 66 characters, for a total of 194 characters. For a FOCUS data source, file names can be up to 64 characters, segment names up to 8 characters, and field names up to 66 characters, for a total of 138 characters.

Because of the two QUALCHARs included with every long qualified field name, the total length of a long qualified field name, usually displayed as two dots, can be up to 196 characters for a relational or an XFOCUS data source, and up to 140 characters for a FOCUS data source.

To use long qualified field names, FIELDNAME must be set to NEW. Long field names can be coded in a Master File or in a DEFINE command. They can be used on z/OS and OS/390, UNIX, or Windows NT.

Example: Using Long Qualified Field Names

Sample Master File:

```
FILENAME = INTSALES,SUFFIX = FOC
SEGNAME = FRANCE,SEGTYPE = S1
FIELDNAME = PROVINCE_NICE_STORE500_TOTALSALES, TSALE500, A10 ,$
FIELDNAME = PROVINCE_NICE_STORE501_TOTALSALES, TSALE501, A10 ,$
```

Sample FOCEXEC:

Note: For display purposes, the FOCEXEC contains CREATE FILE and MODIFY commands.

```
DYNAM ALLOC DD INTSALES LONGNAME -
 AVANT_GARDE_CLOTHING_DATASOURCE -
 DS PMSESD.INTSALES.FOCUS SHR REU
-RUN
-*
CREATE FILE
AVANT_GARDE_CLOTHING_DATASOURCE
-RUN
MODIFY
 FILE AVANT_GARDE_CLOTHING_DATASOURCE
 FREEFORM TSALE500 TSALE501
 MATCH TSALE500
 ON MATCH REJECT
 ON NOMATCH INCLUDE
DATA
 12235,5470,$
 9827,7675,$
 7635,9327,$
10325,7543,$
END
-RUN
TABLE FILE AVANT_GARDE_CLOTHING_DATASOURCE
 PRINT
 AVANT_GARDE_CLOTHING_DATASOURCE.FRANCE.PROVINCE_NICE_STORE500_TOTALSALES
 AVANT_GARDE_CLOTHING_DATASOURCE.FRANCE.PROVINCE_NICE_STORE501_TOTALSALES
END
-RUN
```

The output is:

```
  PAGE     1


  PROVINCE_NICE_STORE500_TOTALSALES  PROVINCE_NICE_STORE501_TOTALSALES
  ---------------------------------  ---------------------------------
  12235                              5470
  9827                               7675
  7635                               9327
  10325                              7543
```

Size Restrictions for Generated Alpha Fields

To restrict the size of alpha fields created by the CREATE SYNONYM command for long character fields, use the following SET command

```
ENGINE engine SET CONVERSION LONGCHAR number
```

where:

engine

Is a valid adapter tag (suffix).

number

Is any number from 1 to 32767. This restricts the size of An and AnV fields created by the CREATE SYNONYM command. 32767 is the default value.

For a Full Function Server application, the recommended values are up to 4096.

The setting also controls the size of answer set columns created by Direct SQL Passthru.

SQL Join Optimization

The heterogeneous joins between two DBMSs or between the data sources on two sub-servers is enhanced in the following manner:

- Predicate of EQ and LIKE is cloned from one table to another in the join.
- Equi-joins order of retrieval is now based on the size (cardinality) of the table. Smaller tables are read first.

The cardinality may be obtained by using a new option of the CREATE SYNONYM command: WITH STATISTICS.

Text Fields for Relational Data Adapters

Text (TX) type fields have been enhanced to support:

- Text strings up to 32K.
- The WHERE optimization clause.

32K Text Strings for Text Fields

Columns described with ACTUAL = TX in a Master File support text strings up to 32K—for example, DB2 CLOB and LONG VARCHAR columns. These columns can be used for read-only operations.

Syntax: How to Describe a Text Field in the Master File

```
FIELDNAME = fname, ALIAS = sqlcolname, USAGE = TXnn, ACTUAL = TX ,$
```

where:

fname

Is any valid field name.

sqlcolname

Is the RDBMS column name.

nn

Is the length of an output line for display. The maximum line length is 32K.

Example: Describing a DB2 CLOB Column

In the Master File:

```
FIELD = CLOB_COL, ALIAS = CLOB_CO, USAGE = TX100, ACTUAL = TX, MISSING = ON,$
```

A request that retrieves this column can display text output up to 32K in length.

WHERE Optimization Clause for Text Fields

Example:

Optimizing a Selection Test Against a Variable Length Character Column

Reference:

Usage Notes for Optimization of Selection Criteria Using Variable Length Data Types

When creating a table request, WHERE criteria can reference RDBMS variable character data types such as VARCHAR, LONG VARCHAR, and CLOB (described in the Master File with USAGE = TX and ACTUAL = TX).

In addition, certain types of IF and WHERE criteria that reference variable length character data types are included in the generated SQL, causing the selection operations to be performed by the RDBMS and improving performance.

The IF or WHERE test to be optimized must be a CONTAINS or OMITS test against a field described with USAGE = TX and ACTUAL = TX in the Master File. The RDBMS column must be a character variable length data type.

CONTAINS translates to LIKE in the generated SQL, and OMITS translates to NOT LIKE. The generated SQL places wildcard characters around the literal string specified in the CONTAINS or OMITS test.

Example: Optimizing a Selection Test Against a Variable Length Character Column

Consider the following variation of the DB2 Master File named EMPINFO. A CLOB column that contains a job description has been added:

```
FILENAME = EMPINFO ,SUFFIX = SQLDS ,$

SEGNAME = EMPINFO ,SEGTYPE = S0 ,$
FIELD = EMP_ID ,ALIAS = EID ,USAGE = A9 ,ACTUAL = A9 ,$
FIELD = LAST_NAME ,ALIAS = LN ,USAGE = A15 ,ACTUAL = A15 ,$
FIELD = FIRST_NAME ,ALIAS = FN ,USAGE = A10 ,ACTUAL = A10 ,$
FIELD = HIRE_DATE ,ALIAS = HDT ,USAGE = YMD ,ACTUAL = DATE ,$
FIELD = DEPARTMENT ,ALIAS = DPT ,USAGE = A10 ,ACTUAL = A10, MISSING = ON ,$
FIELD = CURRENT_SALARY,ALIAS = CSAL ,USAGE = P9.2 ,ACTUAL = P4 ,$
FIELD = CURR_JOBCODE ,ALIAS = CJC ,USAGE = A3 ,ACTUAL = A3 ,$
FIELD = JOBDESC ,ALIAS = JDSC ,USAGE = TX50 ,ACTUAL = TX ,$
FIELD = ED_HRS ,ALIAS = OJT ,USAGE = F6.2 ,ACTUAL = F4,
 MISSING = ON,$FIELD = BONUS_PLAN ,ALIAS = BONUS_PLAN ,USAGE = I4, ACTUAL = I4,$
```

The following request specifies a CONTAINS test against the JOBDESC field:

```
SET TRACEUSER = ON
SET TRACEOFF = ALL
SET TRACEON = STMTRACE//CLIENT

TABLE FILE EMPINFO
PRINT EMP_ID LAST_NAME FIRST_NAME DEPARTMENT CURR_JOBCODE JOBDESC
WHERE JOBDESC CONTAINS 'PR'
END
```

The CONTAINS operator is translated to a LIKE operator in the generated SQL:

```
SELECT T1."EID",T1."LN",T1."FN",T1."DPT",T1."CJC",T1."JDSC"
FROM USER1."EMPINFO" T1 WHERE (T1."JDSC" LIKE '%PR%') FOR
FETCH ONLY;
```

Reference: Usage Notes for Optimization of Selection Criteria Using Variable Length Data Types

The following options are not yet supported with text fields:

- CRTFORM
- TYPE
- FSCAN
- MODIFY
- HOLD
- DEFINE
- COMPUTE

SUBSTRING Function Behavior

In Release 5.2.3, you can choose to retain the SUBSTRING function behavior from previous releases.

Syntax: How to Specify SUBSTRING Function Behavior

```
SQL SET SUBSTRING={ANSI|OLD}
```

where:

ANSI

Returns the SUBSTRING in ANSI VARCHAR format. ANSI is the default value.

OLD

Returns the SUBSTRING in CHAR format.

VARCHAR Support

The TX and CLOB fields are merged to simplify the task of selecting the format:

- A (CHAR) fixed length of n <= 32k. This format supports full DML/SQL operations.
- AnV (VARCHAR) with maximum of n <= 32k. This format supports full DML/SQL operation according to ANSI.
- TX unlimited VARCHAR (< 32k for a FOCUS data source). This format supports limited DML operations, as well as limited SQL operations, which require field level fetch.

CHAPTER 9

WebFOCUS Maintain Enhancements

This chapter describes all of the features for WebFOCUS Maintain in Version 5 Release 2 and has been updated to reflect Version 5 Release 2.3. Topics that contain Version 5 Release 2.3 features are marked with asterisks.

Topics:

General Enhancements

Tighter Integration With Developer Studio

Support for Standard WebFOCUS CGI Aliases

Access to the Installation Verification Program

Performance Enhancements

Wizards, Editors, and Other Tools

* Deploying Only Changed Files With Smart Deploy

Running Components From the Developer Launch Console

Editing Master Files as Text

* Customizing Parsing Activity

* Searching for Text in a Project

Creating Applications With Update Assist

Forms Development

* Applying a Cascading Style Sheet to a Form

Importing a Form Into a Procedure

Dynamically Managing the Display of Controls

Preselecting List Items by Value

New Sample Launch Page

Debugging

Multiple-Level Tracing With Statement Trace

Application-Level Tracing With TYPE ON EDAPRINT

Procedures and Script Development

Retrieving WebFOCUS CGI Parameters

Importing and Binding Variables From a WebFOCUS Report

Changing WebFOCUS Settings From a Procedure

Issuing SQL Commands in a Procedure

Examining RDBMS Return Codes in a Procedure

WebFOCUS Maintain Functions

Enhanced Screening Options

File Designations for Procedures

* Testing Procedures With Run Procedure

Syntax Parsing in Scripts

Resource and Data Access

* Specifying Additional Search Paths

Support for Data Source Access Passwords

Coding Auto-Increment Columns

Concurrent FOCUS Data Source Access

Variable Varchar Data Type Support

Integration With Other Products

* Additional Microsoft Access Support

* WebFOCUS Connector for Excel

Tighter Integration With Developer Studio

The Maintain Development Environment has been integrated more tightly with WebFOCUS Developer Studio. The following features apply:

- The Maintain Development Environment uses Developer Studio for the creation and management of *projects* (what the development versions of applications are now called). You can create a project in the Maintain Development Environment, as part of Developer Studio.

  For a full description of the Explorer interface, from which you access all development tools and facilities, see the *Exploring Your WebFOCUS Development Environment* chapter in the *Developer Studio Application Development Getting Started* manual.

- WebFOCUS Developer Studio and WebFOCUS Maintain share the same deployment tool, which has been completely redesigned for maximum ease of use.

  For complete instructions on taking a project live using the Deploy Wizard, see the *Partitioning and Deploying Project Files* chapter in the *Developing Reporting Applications* manual.

- The currently running Maintain procedure is displayed on the WebFOCUS Reporting Server Console in the Last Command column in the agent and session windows.

  For a full description of these windows, see *Monitoring Agents* and *Monitoring Sessions* in the *Running and Monitoring Your Server* chapter in the *iWay Server Administration for UNIX, Windows, OpenVMS, OS/400, OS/390, and z/OS* manual.

Support for Standard WebFOCUS CGI Aliases

WebFOCUS Maintain uses the standard WebFOCUS CGI aliases (virtual directories), including \approot, \ibi_html, and \cgi-bin\ibi_cgi. This means that WebFOCUS Maintain and WebFOCUS Reporting are more tightly integrated:

- The WebFOCUS CGI automatically adapts to the function call IBIF_*command* to execute procedures using the standard WebFOCUS Maintain command set (for example, MNTCON EX and MNTCON RUN).
- Maintain can take advantage of WebFOCUS CGI scripting. Scripting is fully documented in the *WebFOCUS Security and Administration* manual. For specific information on the use of WebFOCUS script commands and their syntax, see *WebFOCUS Script Commands* in the *WebFOCUS Client Administration Using Additional Customization Techniques* chapter in the *WebFOCUS Security and Administration* manual.

You no longer need to configure special CGI aliases to support standard and new WebFOCUS Maintain applications. If you migrate legacy applications, make sure to change any hard-coded paths to use the new aliases.

To support certain legacy WebFOCUS Maintain applications, you may still need to configure ibi_start and ibi_cgi as aliases from the Web server configuration window.

Aliases are fully documented in the *Configuring the Web Server* chapter in the *WebFOCUS and ReportCaster Installation and Configuration* manual for your platform.

Access to the Installation Verification Program

You can run the WebFOCUS Maintain Installation Verification Program (MNTIVP.MNT) from the sample launch page to confirm the success of a Maintain installation.

The sample launch page is named wfmstart.html. It is located in the /approot/ibisamp/ folder on the Web server in the WebFOCUS environment.

The installation procedure is fully documented in the *WebFOCUS and ReportCaster Installation and Configuration* manual for your platform.

Performance Enhancements

Performance enhancements affect execution and memory:

- Execution time for Web transaction screens has been optimized by 50%.
- Memory usage has been optimized in a number of areas:
  - Compiled procedures take 30% less memory on average.
  - User stacks start with a smaller initial size.
  - WEBFORMs use less memory.

Deploying Only Changed Files With Smart Deploy

How to:

Create a New Deployment Scenario

Edit an Existing Deployment Scenario's Properties

In Release 5.2.3, Smart Deploy has been incorporated in the new deployment architecture (Deploy Wizard). Smart Deploy can significantly reduce the time required to deploy an application because it deploys only files that changed since the last deployment.

When Smart Deploy is active, after the first successful deployment, date/time information is stored in the project's GFA (Graphical FOCUS Application) file for each file deployed. Each subsequent deploy action compares the date/time stamp of the source files in the project against the date/time stamps stored in the deployment scenario. With Smart Deploy, only newer files (source files with a later time stamp than that stored in the GFA) are deployed to the destination.

The date/time information is stored uniquely in each deployment scenario. If a project has multiple deployment scenarios, there is no ambiguity about the date/time the files were last deployed using a specific scenario. When you first use a new deployment scenario, Smart Deploy performs a full deployment of all files referenced for deployment in the project; subsequently, only changed files are deployed.

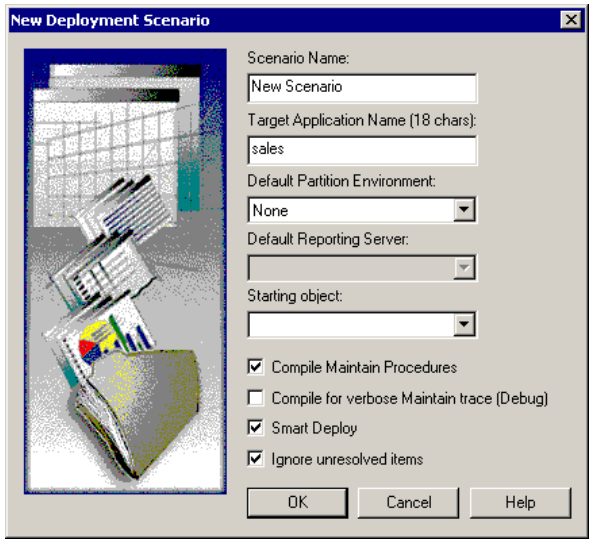
Smart Deploy stores date/time stamps in the GFA after a successful deployment. If an error occurs and the scenario was previously deployed, the date/time stamps are rolled back to the last deployed date. If you add a new file for deployment, the GFA has no date/time stamp for that file, so it is always deployed the first time.

Make sure the Smart Deploy option is checked for the deployment scenario. Do this when creating a new deployment scenario, or by editing an existing deployment scenario's properties.

Local deployment automatically applies Smart Deploy logic, maximizing the speed of the development and testing cycle. All other deployment scenarios default to Smart Deploy.

Procedure: How to Create a New Deployment Scenario

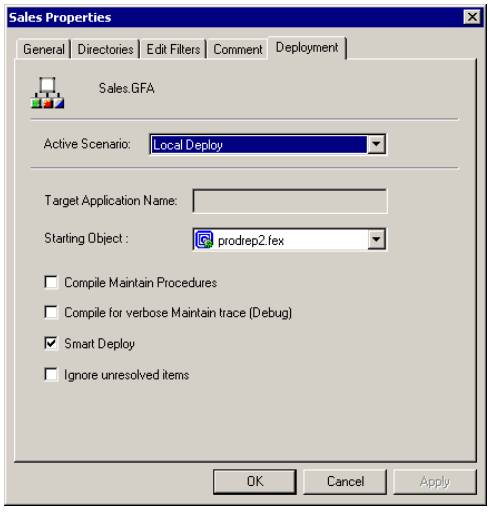
1. Select and right-click the project. Select *Project Deployment* from the shortcut menu, then select *New Scenario*.
2. The New Deployment Scenario dialog box opens. Confirm that Smart Deploy is checked.



For complete instructions on creating a deployment scenario, see *Step 2. Create a Deployment Scenario* in the *Partitioning and Deploying Project Files* chapter in the *Developing Reporting Applications* manual.

Procedure: How to Edit an Existing Deployment Scenario's Properties

1. Select and right-click the project. Select *Properties* from the shortcut menu.
2. On the Project Properties dialog box, select the *Deployment* tab. Confirm that Smart Deploy is checked.



Running Components From the Developer Launch Console

The Developer Launch Console enables you to view applications and execute WebFOCUS procedures and forms.

The Developer Launch Console presents a tree view from which you select components to launch. For example, you can launch WebFOCUS Maintain and WebFOCUS Reporting procedures and open HTML forms on the Web server.

Procedure: How to Run Components From the Developer Launch Console

1. Start the WebFOCUS Server. Enter the following URL in the browser

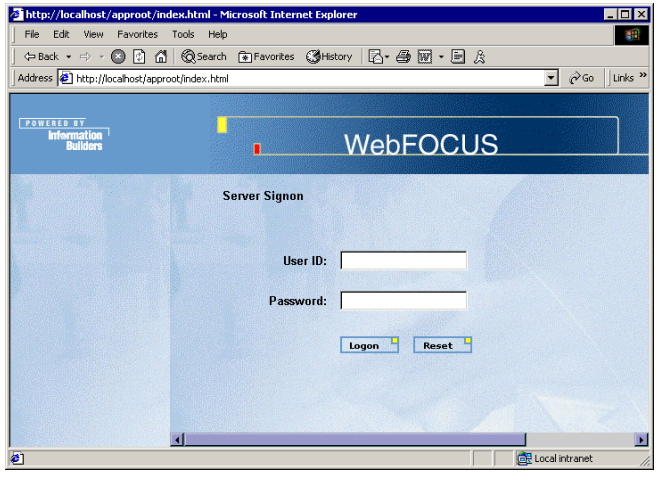
   `http://localhost/ibi_html/applaunch.html`

   where:

   localhost

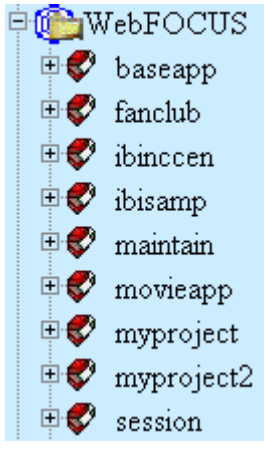
   Is the Web server running the WebFOCUS Client, or your local development server.

2. Sign on to the WebFOCUS Server.

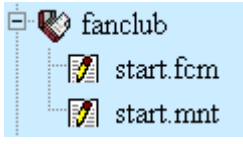


   After signing on, you see in the left pane a list of WebFOCUS Servers available to you.

3. If you expand WebFOCUS or WebFOCUS DATA SERVERS under any WebFOCUS Server, you see a list of the applications on that server.



4. The WebFOCUS folders contain the HTML forms for each application. The WebFOCUS DATA SERVERS folders contain the WebFOCUS and Maintain procedures. If you expand an application, you see the procedures or forms in that application.



5. To run a procedure or HTML form, click it. The results appear in the pane on the right.

Editing Master Files as Text

You can edit a Master File as text in the Maintain Development Environment. To do this, right-click any Master File, and click *Edit As Text*.

Customizing Parsing Activity

In Release 5.2.3, you can control how long the parser waits for you to stop typing before it automatically parses the Maintain code. You can also disable the parser completely to optimize the Maintain Development Environment when you are in a typing-intensive session. You can reenable the parser to update the Project Explorer tree automatically.

Procedure: How to Customize Parsing Activity

1. To change or adjust the parsing option for the Maintain Development Environment (MDE), select *Tools* > *Environment options*. WebFOCUS Maintain displays the Maintain Development Environment Options dialog box.
2. On the General tab, check *Automatically parse editor* to enable parsing of your code in the code Editor, or uncheck it to disable code parsing.

   If the parser is enabled, it waits until you stop typing before parsing your code for changes that might affect the Project Explorer.

   If you disable parsing, changes to your code are not reflected in the Project Explorer until you save them.
3. You can change the amount of time the parser waits before parsing your code.

   Use the spin button to adjust the wait time (in seconds) or type the desired wait time in the designated field.

Searching for Text in a Project

In Release 5.2.3, the search project feature in WebFOCUS Maintain uses the CGI or ISAPI to find requested instances of text in a project. It returns a list of matching text sourced as XML, showing the component name in which the text occurs and the text.

Procedure: How to Search for Text in a Project

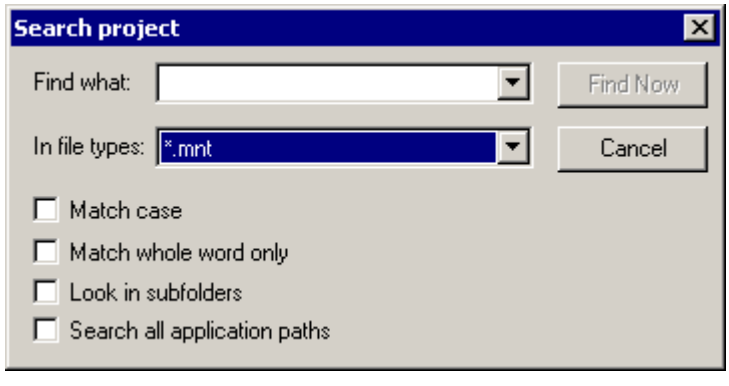
1. In the Edit menu, click *Search project*.

   or

   Click the *Search project* button on the General toolbar.

2. In the Search project dialog box, enter the text to search for in the Find what field, or use the text from a previous search by selecting it from the drop-down list.

   Select the file type from the In file types drop-down list, or select *.* to search all file types.



3. Check the options to apply to the search: Match case, Match whole word only, Look in subfolders, or Search all application paths.
4. Click *Find Now*.
5. If the Output window is not open, open it by clicking *Output window* in the View menu.
6. WebFOCUS Maintain displays a list of the project components that contain the text in the Find tab of the Output Window. To open a component at an instance of the text, double-click its line in the Find tab.

Creating Applications With Update Assist

Update Assist allows you to create fully functional update applications by answering a few simple questions. It is ideal for generating prototypes of WebFOCUS Maintain-based applications.

For a description of Update Assist features, see *Update Assist* in Chapter 2, *Developer Studio Enhancements*.

Applying a Cascading Style Sheet to a Form

How to:

Apply a Cascading Style Sheet to a Form

Apply a Class to a Control on a Form

In Release 5.2.3, you can use an external Cascading Style Sheet (CSS) when developing a WebFOCUS Maintain form. A Cascading Style Sheet enables you to format the appearance of forms using one separate file, rather than having formatting information appear throughout all your files. It is a standard means of separating look from structure.

The formatting information for a form or control comes from one of three places, in the following order of precedence:

1. You can set its value explicitly in Maintain code. This value takes precedence over any other setting.
2. You can apply a Cascading Style Sheet to the form and apply a class to any control on the form. The control uses the characteristics of the class defined in the CSS.
3. You can apply a Cascading Style Sheet to the form (but not apply any classes to the controls). The form and the controls use the characteristics defined in the CSS.

You can use all of these types of styling in the same Maintain form, allowing for powerful and flexible styling capabilities.

Cascading Style Sheets are fully documented on the World Wide Web Consortium (W3C) Web site.

Syntax: How to Apply a Cascading Style Sheet to a Form

Include the following command in the procedure

```
WINFORM SET form.CSSname TO 'stylesheet';
```

where:

form

Is the name of the form to which you are applying the CSS.

stylesheet

Is the name of the CSS. It can be the name of a CSS file in the application path, a relative addressed file name (*path**path**file*), or an HTTP qualified file name.

The form and all the controls on the form with properties set to Default use the settings from the CSS.

Syntax: How to Apply a Class to a Control on a Form

Assuming you have applied a CSS to a form, you can apply a class to any control on the form using the following command

```
WINFORM SET form.[control.]CSSClass TO 'classname';
```

where:

form

Is the name of the form to which you applied the CSS.

control

Optionally, is the name of a control on the form.

classname

Is the name of the class.

If the class name is set to a value other than a blank, WebFOCUS Maintain explicitly sets the name to the supplied value. If the class name is set to blank, the WebFOCUS Server does not write a class name to the control's attributes (which means the control is formatted using the most specific default selector for the control type or form, for example, BODY, INPUT, TEXTAREA, and so on). If you do not set the class name in this manner, Maintain uses the legacy class name for the control (IWC*controlname*).

Formatting for the control inherits the characteristics of the class as specified in the CSS. All the visual properties of the control that are set to Default use the settings for that class in the CSS. Any format overrides set in the Property sheet for a control are applied directly to that control using a style tag in its attributes.

Importing a Form Into a Procedure

How to:

Export a Form

Import a Form

In previous releases of WebFOCUS Maintain, forms were sourced inline inside a Maintain-only file format called CTA (Cactus Application). This feature allowed you to transfer projects between machines, but individual components were not transferable between files. The project control file (GFA) does not contain individual components, only references to them. Forms you design in WebFOCUS Maintain are sourced inside the related Maintain procedure. Forms are sourced in XML and are converted to WINFORM code when you run a project from the Maintain Development Environment or deploy it to a remote server.

The developer's tool set gives you the ability to export any WebFOCUS Maintain form to a file with the extension .for. You can then import the .for file into another Maintain project. This feature gives you the ability to create and reuse form templates from project to project, and allows you to share form templates with other developers.

When you import a form from one procedure to another, WebFOCUS Maintain preserves bindings to resources such as Web links, scripts, and graphics, but not to variables or data source stacks, since the latter are defined in the context of a procedure.

Procedure: How to Export a Form

1. In the Project Editor, right-click the form and click *Export*.
2. In the Export As dialog box, enter a name for the form and click *Save*.

   WebFOCUS Maintain saves the form in a file with the extension .for in the project folder.

Procedure: How to Import a Form

1. In the Project Editor, right-click the Maintain procedure you want to import the form into and click *Import forms*.
2. In the Import Form dialog box, select the form you want to import and click *Open*.

   If you do not see the form you want to import, make sure that it is in one of your project paths.

Dynamically Managing the Display of Controls

You can dynamically manage the visible property of a control through JavaScript or VBScript.

Syntax: How to Dynamically Manage the Display of a Control

```
document.Formname.Controlname.style.visible = "{true|false}";
```

where:

Formname

Is the name of the form.

Controlname

Is the name of the control.

`true`

Enables the display of a control.

`false`

Suppresses the display of a control.

Preselecting List Items by Value

How to:

Preselect List Items by Value

Example:

Preselecting List Items by Value

To preselect list items by value, you can create a stack and search it for the matching items. However, if you know the values to preselect (typically found in data source fields) and want to write less code, use the SelectedItems property.

This feature allows you to preselect one or more items in a multi-select list box by value from Maintain code. You can call it from any event handler.

Syntax: How to Preselect List Items by Value

```
WINFORM SHOW_INACTIVE Formname;
Formname.ListBoxName.SelectedItems(1) = "value1";
Formname.ListBoxName.SelectedItems(2) = "value2";
.
.
.
Formname.ListBoxName.SelectedItems(m) = "valuen";
```

where:

Formname

Is the name of the form.

ListBoxName

Is the name of the list box.

ListItems

Is the name of an internal data source stack that contains the values for the list box.

SelectedItems

Is the name of an internal data source stack that can contain an array of multiple preselected items. Like all stacks, it is a 1-based, not 0-based, array.

value1, value2, valuen

Is a preselected item.

Example: Preselecting List Items by Value

```
Form1.ListBox1.SelectedItems(1) = "NY Office";
```

To select multiple items, pass each value on a separate line:

```
Form1.ListBox1.SelectedItems(1) = "NY Office";
Form1.ListBox1.SelectedItems(2) = "Chicago Office";
Form1.ListBox1.SelectedItems(3) = "LA Office";
```

New Sample Launch Page

The wfmstart.html launch page offers the functionality of the wfmpriv1.htm page from previous releases, but supports the common WebFOCUS CGI calling protocols and APP ENABLE setting that are the global defaults in the Release 5.2 environment.

The launch page is located in the /approot/ibisamp/ folder on the Web server in the WebFOCUS environment.

This page can launch any WebFOCUS Maintain application. You can also use it as a basis for creating your own launch pages.

Multiple-Level Tracing With Statement Trace

How to:

Set Statement Trace

Trace controls allow application developers to trace the logic flow of WebFOCUS Maintain procedures as well as measure CPU and memory used at the procedure, case, and statement level. First compile the procedure to provide the desired level of tracing:

- MNTCON COMPILE *procname* produces trace operation codes for each CASE entry, ENDCASE, CALL, and EXEC statement.

  Tip: A MNTCON EX *procname* does a compile transparently and runs the procedure without writing the compiled version to disk; it therefore also supplies trace information at run time if the SET TRACEON options have been set on the server, as described in *Set Statement Trace* on page 9-15.

- MNTCON DEBUG COMPILE *procname* produces trace operation codes for every WebFOCUS Maintain statement, such as COMPUTE, IF, FOR ALL NEXT.

You can view the trace output from the WebFOCUS Reporting Server Console or by developing reports using the MNTSTMT Master File, which describes the fields used to format the trace information.

For examples of trace output generated by statement trace and additional instructions on the use of this feature, including tracing scenarios, see *Maintain Statement Trace* in the *Debugging WebFOCUS Maintain Applications* chapter in the *Developing WebFOCUS Maintain Applications* manual.

Syntax: How to Set Statement Trace

You must first enable tracing on the server:

- SET TRACEON = MNTSTMT produces basic WebFOCUS Maintain trace information such as file name, case name, line number, and milliseconds.
- SET TRACEON = MNTPERF, used in addition to SET TRACEON = MNTSTMT, generates tracing that includes memory and CPU performance statistics.

Application-Level Tracing With TYPE ON EDAPRINT

Issuing a TYPE ON EDAPRINT directs tracing output to the edaprint.log file, which is viewable from the WebFOCUS Reporting Server Console or SDSF display on OS/390. Use this feature in production applications for communicating tracing information to the application administrator.

Example: Issuing TYPE ON EDAPRINT

The following code

```
MAINTAIN
type on edaprint "first to edaprint";
type "hello world! from user1";
type on edaprint "hello - send to edaprint";
END
```

generates this output in the edaprint.log file:

```
11/20/2002 13:53:28 Pid=00001528 MntType: first to edaprint
11/20/2002 13:53:28 Pid=00001528 MntType: hello - send to edaprint
```

Retrieving WebFOCUS CGI Parameters

How to:

Retrieve a WebFOCUS CGI Parameter

Example:

Retrieving a WebFOCUS CGI Parameter

The WebFOCUS Maintain language allows a Maintain procedure to access WebFOCUS CGI parameters from HTTP request headers or user-defined parameters as issued on typical Web URLs. For example, you can retrieve any parameter passed from a WebFOCUS drill down report to Maintain.

The IWC.GetAppCGIValue function imports the value of a WebFOCUS CGI parameter into a Maintain variable. If the passed CGI parameter name is not found, the function returns a null value.

Tip: Unlike Maintain variables, CGI parameters are case sensitive.

Available WebFOCUS Maintain functions are fully documented in the *WebFOCUS Using Functions* manual.

Syntax: How to Retrieve a WebFOCUS CGI Parameter

```
Declare mnt_var/type_length = IWC.GetAppCGIValue("cgi_parm");
```

where:

mnt_var

Is the Maintain variable that receives the ASCII value of the CGI parameter. The value is unescaped before it is passed to the variable.

type_length

Is the type and length of the Maintain variable.

cgi_parm

Is the CGI parameter, which can be a variable or a literal value enclosed in double quotation marks. The value of *cgi_parm* is case sensitive.

Example: Retrieving a WebFOCUS CGI Parameter

IWC.getAppCGIValue retrieves the value of the CGI parameter PRODUCT_ID:

```
Maintain File GGPRODS
Infer Product_ID into prodstk;
Declare pcode/a4 = IWC.getAppCGIValue("PRODUCT_ID");
For 1 next Product_ID into prodstk where Product_ID eq pcode;
```

Importing and Binding Variables From a WebFOCUS Report

How to:

Import and Bind Variables

Reference:

Variable Binding Guidelines

Import WebFOCUS Parameters Into Maintain Dialog Box

The variable binding feature allows you to bind any Maintain procedures to WebFOCUS drill down reports without having to write code. This feature is especially applicable for users who need to make spot updates from WebFOCUS reports.

To implement this feature, add a drill down to your WebFOCUS report, passing to Maintain all keys needed to locate the record in Maintain. You can then create the WebFOCUS parameter-to-Maintain variable binding without writing Maintain code.

Tip: In WebFOCUS Developer Studio, passed name/value pairs are called *parameters*. In WebFOCUS Maintain, replaceable named values are called *variables*.

Procedure: How to Import and Bind Variables

Before using variable binding in a Maintain procedure, you must create a WebFOCUS report that has a drill down to the Maintain procedure. Once you have created the report and specified the drill down parameters, do the following:

1. In the Maintain Development Environment, open the Maintain procedure in the Procedure Editor.
2. Place your cursor in the $$Declarations section.
3. Right-click and select *Import WebFOCUS Parameters*.
4. In the Open dialog box, select the WebFOCUS report you want to import parameters from and click *OK*.
5. In the Import WebFOCUS Parameters into Maintain dialog box, assign new data types to the parameters you want to import using the Type Wizard and click *OK*.

   WebFOCUS Maintain places the following code at the beginning of the procedure:

   ```
   Declare variable/A4 = IWC.GetAppCgiValue("variable");
   ```

Reference: Variable Binding Guidelines

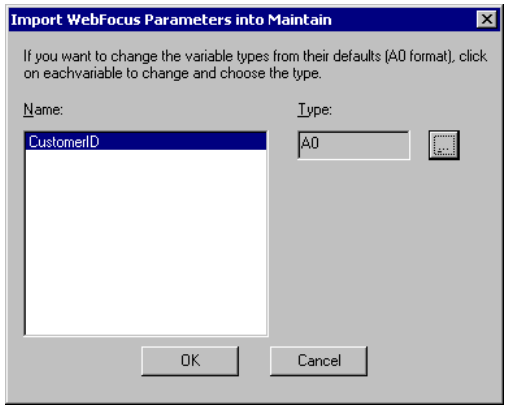
- If you import a parameter from a WebFOCUS report into an existing Maintain procedure, make sure that the parameter name matches the variable name used in the procedure.

  For example, if you use a variable named CustomerNo in a Maintain procedure to determine which record is pulled from a customer data source, but you use CustomerID as the parameter name in the WebFOCUS report that launches the Maintain procedure, variable binding will not work.

  You can, however, manually change the variable name in the Maintain procedure so that it matches the parameter name.
- You cannot import parameters from a WebFOCUS report that drills down to a different Maintain procedure.

Reference: Import WebFOCUS Parameters Into Maintain Dialog Box

When importing parameters from a WebFOCUS report into a Maintain procedure, use the Import WebFOCUS Parameters into Maintain dialog box to change the format of the parameters.


Import WebFocus Parameters into Maintain
If you want to change the variable types from their defaults (A0 format), click on eachvariable to change and choose the type.
Name:
Type:
CustomerID
A0
OK
Cancel

This dialog box has the following fields:

Name: Contains the names of the parameters from the WebFOCUS report imported into the Maintain procedure.

Type: Contains the proposed format for a parameter.

Opens the Type Wizard, from which you can specify a new data type for the parameter.

Changing WebFOCUS Settings From a Procedure

How to:

Change WebFOCUS Settings From a Procedure

Set CDN From a Procedure

Set COMMIT From a Procedure

Set DATEDISPLAY From a Procedure

Set DEFCENT From a Procedure

Set EMGSRV From a Procedure

Set LANGUAGE From a Procedure

Set MESSAGE From a Procedure

Set NODATA From a Procedure

Set PASS From a Procedure

Set TRACEOFF From a Procedure

Set TRACEON From a Procedure

Set TRACEUSER From a Procedure

Set USER From a Procedure

Set WARNING From a Procedure

Set YRTHRESH From a Procedure

Example:

Setting DEFCENT From a Procedure

Setting PASS From a Procedure

Enabling Server Tracing for MNTSTMT

The SYS_MGR.FOCSET function allows you to change certain WebFOCUS environment, server, and other settings from a Maintain procedure. This feature gives greater flexibility and control within a Maintain session.

Syntax: How to Change WebFOCUS Settings From a Procedure

Tip: Single quotation marks are allowed in place of double quotation marks.

```
SYS_MGR.FOCSET("parm","value");
```

where:

parm

Is one of the supported SET parameters:

```
CDN
COMMIT
DATEDISPLAY
DEFCENT
EMGSRV
LANGUAGE
MESSAGE
NODATA
PASS
TRACEOFF
TRACEON
TRACEUSER
USER
WARNING
YRTHRESH
```

value

Is a valid setting for the parameter. See the Syntax topics for the default value and valid settings for each parameter.

Syntax: How to Set CDN From a Procedure

CDN (Continental Decimal Notation) determines the punctuation used in numerical notation. When the CDN setting is ON, a comma marks the decimal position, and periods separate groups of three significant digits.

```
SYS_MGR.FOCSET("CDN","option");
```

where:

option

Determines the punctuation used in numerical notation. Possible values are:

ON enables CDN. For example, the number 3,045,000.76 is represented as 3.045.000,76.

OFF disables CDN. For example, the number 3,045,000.76 is represented as 3,045,000.76. OFF is the default value.

SPACE separates groups of three significant digits with a space instead of a comma, and marks a decimal position with a comma instead of a period. For example, the number 3,045,000.76 is represented as 3 045 000,76.

QUOTE separates groups of three significant digits with a single quotation mark instead of a comma, and marks a decimal position with a comma instead of a period. For example, the number 3,045,000.76 is represented as 3'045'000,76.

For further information on the use of CDN and an example, see *Punctuating Numbers* in Chapter 1, *WebFOCUS Reporting Language Enhancements*.

Syntax: How to Set COMMIT From a Procedure

COMMIT enables transaction integrity for FOCUS data sources.

```
SYS_MGR.FOCSET("COMMIT","{ON|OFF}");
```

where:

ON

Enables the COMMIT and ROLLBACK commands for use with FOCUS data sources, and enables the use of the FOCUS Database Server to ensure transaction integrity.

OFF

Disables the COMMIT and ROLLBACK commands for use with FOCUS data sources, and disables the use of the FOCUS Database Server. OFF is the default value.

COMMIT is fully documented in *Ensuring Transaction Integrity for FOCUS Data Sources* in the *Ensuring Transaction Integrity* chapter in the *Developing WebFOCUS Maintain Applications* manual.

Syntax: How to Set DATEDISPLAY From a Procedure

DATEDISPLAY controls the display of a date format field (for example, YYMD) that contains the value zero.

```
SYS_MGR.FOCSET("DATEDISPLAY","{ON|OFF}");
```

where:

`ON`

Returns the base date for a date format field containing the value zero.

`OFF`

Returns a blank for a date format field containing the value zero. OFF is the default value.

Syntax: How to Set DEFCENT From a Procedure

DEFCENT defines a default century globally or on a field level for an application that does not contain an explicit century. DEFCENT is used in conjunction with YRTHRESH to interpret the current century according to the given values. When assigned globally, the time span created by these parameters applies to every 2-digit year used by the application unless you specify file-level or field-level values.

You can achieve the same result by including the FDEFCENT and FYRTHRESH attributes in a Master File.

```
SYS_MGR.FOCSET("DEFCENT","{cc|19}");
```

where:

cc

Is the default century. If you do not supply a value, *cc* defaults to 19, for the twentieth century.

DEFCENT is fully documented in *Working With Cross-Century Dates* in the *Unique FOCUS Topics* category in *Supplementary Documentation* on the documentation CD.

Example: Setting DEFCENT From a Procedure

The following code

```
MAINTAIN
 COMPUTE MYDATE/YYMD;
 SYS_MGR.FOCSET("DEFCENT","21");
 COMPUTE DATE1/YMD = '90/01/01';
 COMPUTE MYDATE = DATE1;
 TYPE "After setting DEFCENT = 21, MYDATE = <<MYDATE";
END
```

produces this output:

```
After setting DEFCENT = 21, MYDATE = 2190/01/01
```

Syntax: How to Set EMGSRV From a Procedure

EMGSRV controls the return of all messages from the WebFOCUS Server.

```
SYS_MGR.FOCSET("EMGSRV","{ON|OFF}”);
```

where:

ON

Returns messages from the WebFOCUS Server. ON is the default value.

OFF

Suppresses messages from the WebFOCUS Server.

Syntax: How to Set LANGUAGE From a Procedure

LANGUAGE specifies the language in which WebFOCUS Server messages in the format FOC*nnn*-FOC*nnnnn* are displayed. The setting applies only to the current Maintain procedure.

```
SYS_MGR.FOCSET("LANG[UAGE]","{value|AMENGLISH}");
```

where:

value

Is a language from the following list. You can use the abbreviation or full language name to specify the language.

Name	Abbreviation	Description
AMENGLISH	AME	Uses the English set of server messages (default).
DUTCH	DUT	Uses the Dutch set of server messages.
FRENCH	FRE	Uses the French set of server messages.
GERMAN	GER	Uses the German set of server messages.
JAPANESE	JPN	Uses the Japanese set of server messages.
SPANISH	SPA	Uses the Spanish set of server messages.

Syntax: How to Set MESSAGE From a Procedure

MESSAGE controls the display of informational messages in the view source window of a Web browser.

```
SYS_MGR.FOCSET("MESSAGE","{ON|OFF}");
```

where:

ON

Displays informational messages. ON is the default value.

OFF

Suppresses both informational messages and carets that appear when WebFOCUS executes commands in procedures. WebFOCUS still displays error messages and the carets that prompt for input.

Syntax: How to Set NODATA From a Procedure

NODATA determines the character string that indicates missing data on a report.

```
SYS_MGR.FOCSET("NODATA","{string|.}");
```

where:

string

Is the character string that indicates missing data. A period (.) is the default value.

Syntax: How to Set PASS From a Procedure

PASS establishes a user's identity for WebFOCUS DBA. A user must enter the password before accessing any data source that has WebFOCUS DBA security applied to it.

When setting a password for DBA access, keep in mind that the last value set from the application is in effect for all transactions for that end user's session.

```
SYS_MGR.FOCSET("{PASS|USER}","name");
```

where:

name

Is the user's name or password.

The use of this parameter with encryption and decryption is fully documented in *ENCRYPT/DECRYPT* in the *Command Reference* chapter in the *Maintain Language Reference* manual. Implementing password security is fully documented in *Implementing Data Source Security* in the *Providing Data Source Security: DBA* chapter in the *WebFOCUS Security and Administration* manual.

Example: Setting PASS From a Procedure

The following code sets the password to DBAUSER2:

```
SYS_MGR.FOCSET("PASS","DBAUSER2");
```

Syntax: How to Set TRACEOFF From a Procedure

TRACEOFF enables you to turn off all or specific trace options. Values are available to suppress different kinds of server tracing for Information Builders products. In this syntax, Maintain-specific values are shown.

```
SYS_MGR.FOCSET("TRACEOFF","{ALL|MNTSTMT|MNTPERF}");
```

where:

ALL

Disables all available trace options. ALL is the default value.

MNTSTMT

Disables basic WebFOCUS Maintain trace information such as file name, case name, line number, and milliseconds.

MNTPERF

Disables trace information that includes memory and CPU performance statistics.

Syntax: How to Set TRACEON From a Procedure

TRACEON controls the level and scope of trace information. Values are available to generate different kinds of server tracing for Information Builders products. In this syntax, Maintain-specific values are shown.

```
SYS_MGR.FOCSET("TRACEON","{ALL|MNTSTMT|MNTPERF}");
```

where:

ALL

Enables all available trace options. ALL is the default value.

MNTSTMT

Generates basic WebFOCUS Maintain trace information such as file name, case name, line number, and milliseconds.

MNTPERF

Used with MNTSTMT, generates trace information that includes memory and CPU performance statistics.

Syntax: How to Set TRACEUSER From a Procedure

TRACEUSER controls trace logging.

```
SYS_MGR.FOCSET("TRACEUSER","{ON|OFF}");
```

where:

`ON`

Starts or resumes trace logging.

`OFF`

Stops trace logging. OFF is the default value.

Example: Enabling Server Tracing for MNTSTMT

The following commands turn on trace logging, disable all trace options, then turn on tracing for statement trace only (MNTSTMT):

```
SYS_MGR.FOCSET("TRACEUSER","ON");
SYS_MGR.FOCSET("TRACEOFF","ALL");
SYS_MGR.FOCSET("TRACEON","MNTSTMT");
```

Syntax: How to Set USER From a Procedure

USER establishes user access to a data source protected by Information Builders security. For syntax, see *Set PASS From a Procedure* on page 9-26.

Syntax: How to Set WARNING From a Procedure

WARNING controls the return of warning messages from the WebFOCUS Server.

```
SYS_MGR.FOCSET("WARNING","{ON|OFF}");
```

where:

`ON`

Returns warning messages from the WebFOCUS Server. ON is the default value.

`OFF`

Suppresses warning messages from the WebFOCUS Server.

Syntax: How to Set YRTHRESH From a Procedure

YRTHRESH defines the start of a 100-year window globally or on a field level. Used with DEFCENT, it interprets the current century according to the given values. Two-digit years greater than or equal to YRTHRESH assume the value of the default century. Two-digit years less than YRTHRESH assume the value of one more than the default century.

The same result can be achieved by including the FDEFCENT and FYRTHRESH attributes in a Master File.

```
SYS_MGR.FOCSET("YRTHRESH","{yy|0}");
```

where:

yy

Is the year threshold for the window. If you do not supply a value, *yy* defaults to zero (0).

If *yy* is a positive number, that number is the start of the 100-year window. Any two-digit years greater than or equal to the threshold assume the value of the default century. Two-digit years less than the threshold assume the value of one more than the default century.

If *yy* is a negative number (*-yy*), the start date of the window is derived by subtracting that number from the current year, and the default century is automatically calculated. The start date is automatically incremented by one at the beginning of each successive year.

YRTHRESH is fully documented in *Working With Cross-Century Dates* in the *Unique FOCUS Topics* category in *Supplementary Documentation* on the documentation CD.

Issuing SQL Commands in a Procedure

How to:

Issue an SQL Command in a Procedure

Example:

Inserting a Row Into a Table (MS-SQL)

Dropping the Table NYACCTS (MS-SQL)

Setting Connection Attributes for an MS-SQL Server

Passing Credentials to a Remote Server

You can pass native SQL commands and access parameters directly to an RDBMS from a WebFOCUS Maintain procedure without translation or syntax checking. For example, applications can create and drop tables, create indexes, and even make Remote Procedure Calls (RPCs).

Use DBMS_ERRORCODE to examine the RDBMS return code to determine the success or failure of a command. For more information on DBMS_ERRORCODE, see *Examining RDBMS Return Codes in a Procedure* on page 9-32.

Used in conjunction with the retrieval of return codes, this feature gives you greater flexibility and control when designing RDBMS update applications.

You can also send SQL EDA commands for *n*-tier WebFOCUS Maintain applications using FOCUS data sources.

Syntax: How to Issue an SQL Command in a Procedure

```
SYS_MGR.ENGINE("engine","command");
```

where:

engine

Is the name of the RDBMS to which you are passing the command.

command

Is any valid SQL command, including CREATE, INSERT, and DROP.

Example: Inserting a Row Into a Table (MS-SQL)

```
Compute rc/i8;
Type "Inserting row into table MNTBTAB2 ";
rc=sys_mgr.engine("SQLMSS","insert into mntbtab2
values('X2','XDAT2222');");
Type"ReturnCode=<<rc DBMS Err=<<SYS_MGR.DBMS_ERRORCODE";
```

The return code is saved in a variable named rc. Test it to determine whether the record was inserted successfully (rc = 0).

With MS-SQL, if the inserted row is a duplicate, you receive the following return codes:

```
Return Code= -1 DBMS Err= 2627
```

Example: Dropping the Table NYACCTS (MS-SQL)

```
Compute rc/i8;
rc=sys_mgr.engine("SQLMSS","DROP TABLE NYACCTS");
Type "Return Code=<<rc DBMS Err=<<SYS_MGR.DBMS_ERRORCODE";
```

Example: Setting Connection Attributes for an MS-SQL Server

```
Compute rc/i8;
rc=sys_mgr.engine("SQLMSS","set connection_attributes mssxyz/
ibiusr1,mkt");
Type "RC from set is <<rc DBMS Err=<<SYS_MGR.DBMS_ERRORCODE";
```

Example: Passing Credentials to a Remote Server

You can set user and password values for remote WebFOCUS Servers using the SYS_MGR.ENGINE command, and setting *enginename* to EDA. Consequently, you do not need to use EDASPROF to pass the SQL EDA SET USER command.

The following code takes values for the user ID and password from a launch form and passes them to the target server:

```
COMPUTE USER/A8;
PWD/A8;
iwc.findappcgivalue("IBIC_user",USER);
iwc.findappcgivalue("IBIC_pass",PWD);
.
.
.
SYS_MGR.ENGINE("EDA","SET USER EDASERVE/" || USER || ',' || PWD);
EXEC myreport AT EDASERVE into mystack
```

For a remote server requiring different user ID and password values than those supplied at logon, omit the use of the iwc.findappcgivalue function and design the application to gather values from the user at run time.

Examining RDBMS Return Codes in a Procedure

How to:

Examine an RDBMS Return Code in a Procedure

Example:

Coding for Portability

Examining an RDBMS Return Code in a Procedure

The DBMS_ERRORCODE property of the SYS_MGR object is the return code from an RDBMS (the SQLCODE of the last database command). Use DBMS_ERRORCODE to gather more details behind the failure of an operation than are available in FOCERROR.

For example, an INSERT might not complete successfully due to referential integrity failure, lack of permissions, or other reason. DBMS_ERRORCODE contains the specific SQLCODE from the RDBMS so your application can take the correct action.

DBMS_ERRORCODE applies to the current Maintain procedure.

Return codes are RDBMS specific. Also, RDBMS vendors may change return codes between releases. To ensure portability, use a coding technique that is easy to adjust. You should also document the fact that you are using this feature to allow sufficient testing before implementing a new RDBMS.

Syntax: How to Examine an RDBMS Return Code in a Procedure

```
SYS_MGR.DBMS_ERRORCODE;
```

Example: Coding for Portability

```
Compute Ref_Integ_Code/i8 = -843;
.
.
.
Include ProdOrder from OrdStk;
If SYS_MGR.DBMS_ERRORCODE eq Ref_Integ_Code Then...
```

An enhanced technique is to put the calculated field in a common procedure that is imported.

Example: Examining an RDBMS Return Code in a Procedure

The following Maintain code examines a return code from an RDBMS, and depending on its value, branches to the appropriate action:

```
Compute ErrCode/a3 = SYS_MGR.DBMS_ERRORCODE;
If ErrCode EQ '515' goto BadInsert;
```

WebFOCUS Maintain Functions

In this section:

CHAR2INT

INT2CHAR

NLSCHR

GetAppCGIValue

The following functions were introduced in Release 5.2.

CHAR2INT

The CHAR2INT function returns the integer value of an ASCII or EBCDIC character.

Syntax: How to Retrieve the Integer Value of a ASCII or EBCDIC Character

```
CHAR2INT("character")
```

where:

character

Is the ASCII or EBCDIC character.

Example: Retrieving the Integer Value of the Character X

```
MAINTAIN
INT/I3 = CHAR2INT("X");
type "INT IS <INT";
END
```

On an ASCII platform, the returned integer value is 120.

INT2CHAR

The INT2CHAR function returns the character equivalent of a numeric value on a current 256-element code page.

Syntax: How to Retrieve the Character Equivalent of a Numeric Value

```
INT2CHAR(value)
```

where:

value

Is the numeric value on the code page.

Example: Retrieving the Character Equivalent of 93

```
MAINTAIN
CHAR/A1 = INT2CHAR(93);
TYPE "CHAR IS <CHAR";
END
```

On an ASCII platform, the returned character is a right bracket (]).

NLSCHR

The NLSCHR function converts a character on the currently enabled code page to a character on the native English code page. This function is useful when hosting Web applications on an EBCDIC platform with non-English code pages.

Syntax: How to Convert a Character on the Current Code Page

```
NLSCHR("character")
```

where:

character

Is the character being converted.

Example: Displaying a Dollar Sign Regardless of Code Page

NLSCHR forces the display of a dollar sign ($) whenever the variable ADOLLAR is used, regardless of the currently enabled code page.

```
MAINTAIN
ADOLLAR/A1 = NLSCHR("$");
.
.
.
END
```

GetAppCGIValue

The IWC.GetAppCGIValue function imports the value of a WebFOCUS CGI parameter into a Maintain variable. IWC.GetAppCGIValue retrieves the parameter value from the CGI (the HTTP request header).

For syntax and an example, see *Retrieving WebFOCUS CGI Parameters* on page 9-16.

Enhanced Screening Options

In this section:

CONTAINS and OMITS

NOT_IN

EQ_MASK and NE_MASK

The screening options for the WHERE clause in the NEXT command have been enhanced.

Screening options with NEXT are fully documented in *Using Selection Logic to Retrieve Rows* in the *Command Reference* chapter in the *Maintain Language Reference* manual.

CONTAINS and OMITS

You can use CONTAINS and OMITS with a variable, such as a calculated field, in a NEXT command against a data source.

Example: Using CONTAINS With a Calculated Field

```
Compute namepart/a0 = searchname;
   For All Next Employee into Stk Where Empname CONTAINS namepart;
```

NOT_IN

The NOT_IN logical operator selects values that are not in a supplied list.

Example: Selecting Countries Not in a Supplied List

```
For All Next Country into Stk Where Country NOT_IN ("ENGLAND","FRANCE");
```

EQ_MASK and NE_MASK

The EQ_MASK and NE_MASK logical operators compare a data source field to a calculated field that uses $ as a wild card character.

Example: Comparing a Data Source Field to a Calculated Field

```
Compute afld/a8 = "$$$LAND";
   For All Next Country into Stk Where Country EQ_MASK afld;
```

File Designations for Procedures

How to:

Work With Maintain File Designations

Example:

Specifying File Designations

Reference:

Maintain Search Order

Maintain procedures have a new file designation. On portable platforms such as Windows and UNIX, the new extension is .mnt. On OS/390 and z/OS, the ddname is MAINTAIN. On portable platforms, front-end tools generate the extension .mnt for the procedure. On OS/390 or z/OS, you must allocate the ddname MAINTAIN.

Syntax: How to Work With Maintain File Designations

To work with Maintain procedures from a WebFOCUS command line, you must issue the MNTCON command.

To execute a Maintain procedure:

```
MNTCON EX maintproc
```

To compile a Maintain procedure:

```
MNTCON COMPILE maintproc
```

The preceding command creates a compiled procedure with the extension .fcm on Windows and UNIX, or ddname FOCCOMP on OS/390 or z/OS (allocate ddname FOCCOMP).

To run a compiled Maintain procedure

```
MNTCON RUN maintproc
```

where:

maintproc

Is a compiled Maintain procedure (.fcm or FOCCOMP).

Reference: Maintain Search Order

Maintain first searches for a procedure with the extension .mnt or ddname MAINTAIN. If it does not find the procedure with extension .mnt on Windows or UNIX, or with ddname MAINTAIN on OS/390 or z/OS, it searches for extension .fex on Windows and UNIX, or ddname FOCEXEC on OS/390 or z/OS.

Example: Specifying File Designations

In Windows:

```
C:\mydata\test1.mnt
```

In OS/390 JCL:

```
//MAINTAIN DD=myqualif.MAINTAIN.DATA  DISP=SHR
```

Testing Procedures With Run Procedure

In Release 5.2.3, you can test procedures by conducting a local deploy and run using the Run Procedure option.

Procedure: How to Test Procedures With Run Procedure

1. Right-click a procedure in the Object Explorer or Project Explorer window.
2. Select *Run Procedure*.

This option initiates a local deploy and run.

Syntax Parsing in Scripts

When you type JavaScript or VBScript in a script file or an event handler, WebFOCUS Maintain parses the code for syntax errors. Errors are flagged with exclamation marks, as are Maintain syntax errors.

Only valid JavaScript or VBScript is permitted in a script file or event handler. For example, HTML comments (<!-- ... -->) are not valid. Instead, use JavaScript comments (/*...*/).

The use of scripts in Maintain projects is fully documented in *Using Script Functions as Event Handlers* in the *Defining Events and Event Handlers* chapter in the *Developing WebFOCUS Maintain Applications* manual.

Specifying Additional Search Paths

In Release 5.2.3, when issuing a CALL or EXEC statement, use the optional PATH keyword to specify additional locations (search paths) the system should use when searching for dependent resources (Master Files, imported modules, and so on). The path location names are application names existing in the APPROOT directory structure or application names introduced with the APP MAP command.

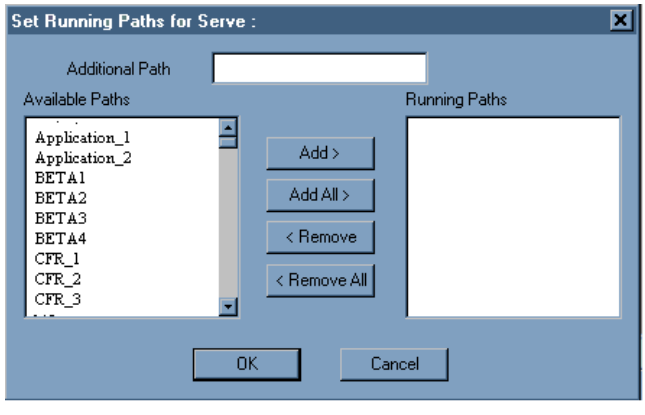
The search path value can be in the form of a Maintain variable, or a list of literal values enclosed in double quotation marks, as follows:

```
EXEC Procedure AT server PATH MyVariable;
CALL Procedure AT server PATH "AppDir1 AppDir2 AppDir3";
```

Procedure: How to Specify an Additional Search Path

The Developer Studio deployment process can introduce the PATH keyword into a CALL or EXEC statement. This is accomplished by the deployment scenario editor, which provides a dialog box that controls the values used for the application path. To access this dialog box:

1. Open the deployment scenario.
2. Right-click the WebFOCUS Environment assigned to the procedure.
3. Select *Paths* from the pop-up menu.



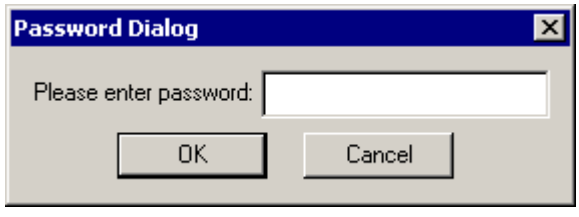
Items included in the Running Paths list appear as PATH values for those procedures that are invoked with the CALL or EXEC statement on the selected WebFOCUS Environment.

Support for Data Source Access Passwords

The Edit menu Password option prompts you to enter the WebFOCUS DBA password required for data source access. WebFOCUS Maintain uses the password to access the Master File.

Procedure: How to Supply a WebFOCUS DBA Password

1. Click *Password* on the Edit menu.
2. Enter a password in the dialog box and click *OK*.



Procedure: How to Collect DBA Information at Run Time

Use a Web form to collect DBA information when your application starts, and to place the DBA password in a special cookie entry called DBAPASS in the WF_USER cookie. You can then modify the WebFOCUS CGI scripts to send DBAPASS to the WebFOCUS Server as a SET PASS = &DBAPASS command in the IBISTART.WFS file. This way, your WebFOCUS Maintain applications can transparently and consistently pass DBA support.

To see this technique, generate an Update Assist application against a DBA-protected Master File, and make sure you select *Prompt user to enter database security information (DBA)* at the Selecting Navigation Options step. The resulting navigation bar illustrates the technique.

For complete instructions on creating an Update Assist application, see the *Creating an Update Application With Update Assist* chapter in the *Developing Reporting Applications* manual.

Coding Auto-Increment Columns

How to:

Specify a Read-Only Field in a Master File

Example:

Using FIELDTYPE = R

Coding an Auto-Increment Column

Auto-increment columns are useful in transaction processing. Use auto-increment columns when many updates could occur for one key field. An example is one social security number with several updates that reflect an individual's multiple purchases.

The field-level Master File attribute FIELDTYPE = R identifies a field as read-only. You can use the read-only attribute for auto-increment columns. This attribute provides support for relational data sources with auto-increment columns such as IDENTIFY or timestamp columns in DB2. These columns are automatically incremented by the RDBMS.

Update languages such as Maintain ignore columns with this designation and allow the RDBMS to handle the automatic generation of the value.

CREATE SYNONYM adds the FIELDTYPE = R attribute for read-only fields in an RDBMS table.

Syntax: How to Specify a Read-Only Field in a Master File

```
FIELD = ..., FIELDTYPE = R,$
FIELD = ..., FIELDTYPE = RI,$
FIELD = ..., FIELDTYPE = IR,$
```

where:

`R`

Indicates that the field is read-only. Any number of fields can have this attribute.

`RI,IR`

Indicates that the field is read-only and indexed (applies to FOCUS, VSAM, and XFOCUS data sources only).

Example: Using FIELDTYPE = R

Autoinc1 is an MS SQL Server table with an IDENTITY field called Control as the first field, which is a key.

The following CREATE SYNONYM command creates a Master File with the FIELDTYPE = R attribute for the Control field:

```
CREATE SYNONYM MNTAUTO FOR AUTOINC1 DBMS SQLMSS AT MSSXYZ
END
```

The generated Master File is:

```
FILE = MNTAUTO , SUFFIX = SQLMSS ,$
SEGNAME = MNTAUTO , SEGTYPE = S0 ,$
FIELD = CONTROL   ,CONTROL   ,I11 ,I4  ,MISSING = OFF, FIELDTYPE = R,$
FIELD = LASTNAME  ,LASTNAME  ,A12 ,A12 ,MISSING = OFF,$
FIELD = FIRSTNAME ,FIRSTNAME ,A12 ,A12 ,MISSING = OFF,$
FIELD = ITEM      ,ITEM      ,A20 ,A20 ,MISSING = OFF,$
FIELD = AMOUNT    ,AMOUNT    ,P19 ,P10 ,MISSING = OFF,$
```

The generated Access File is:

```
SEGNAME = MNTAUTO ,
TABLENAME = AUTOINC1,
CONNECTION = MSSXYZ,
KEYS = 1,$
```

Example: Coding an Auto-Increment Column

The following Maintain procedure, which adds new data, does not assign a value to the auto-increment column called Control or reference it in the INCLUDE command. If a record is successfully included, the RDBMS automatically generates a new value for Control for that record.

```
MAINTAIN FILE MNTAUTO
INFER CONTROL LASTNAME FIRSTNAME ITEM AMOUNT INTO INSTK
PERFORM UPDAT1

CASE UPDAT1
COMPUTE INSTK(1).LASTNAME = 'PIG';
COMPUTE INSTK(1).FIRSTNAME = 'PORKY';
COMPUTE INSTK(1).ITEM = 'PORKBELLIES';
COMPUTE INSTK(1).AMOUNT = 2000.00;
FOR 1 INCLUDE LASTNAME FIRSTNAME ITEM AMOUNT FROM INSTK
ENDCASE

END
```

If the Maintain code were to provide a value for Control in the following commands, the updated value for that field would be ignored:

```
COMPUTE INSTK(1).CONTROL = 1234;
FOR 1 INCLUDE CONTROL LASTNAME FIRSTNAME ITEM AMOUNT FROM INSTK
```

Make sure to query the data source to find out which value was generated by the RDBMS for the auto-increment field.

Concurrent FOCUS Data Source Access

Example:

Setting Up a Shared Application Server For Concurrent Access of FOCUS Data Sources

The Shared Application Server (MNTCON STARTSERVER) supports simultaneous use of FOCUS data sources, and is no longer limited to relational data sources. The Shared Application Server provides the scalability options necessary for the access of WebFOCUS Maintain applications by many concurrent users.

The enhanced Shared Application Server allows the multi-session-per-agent functionality to be used for WebFOCUS Maintain applications updating FOCUS data sources.

For complete instructions on preparing and running an application on a Shared Application Server, see *Developing an Application for a Shared Application Server* in the *Running WebFOCUS Maintain Applications* chapter in the *Developing WebFOCUS Maintain Applications* manual.

Example: Setting Up a Shared Application Server For Concurrent Access of FOCUS Data Sources

The following is a sample profile (EDASPROF.PRF).

To run the starting procedure NYACCTS, use the MNTCON RUNIMAGE NYACCTS command from a launch form.

In the example, the USE command is required for FOCUS data sources (NYACCTS and REGIONS).

```
APP ENABLE
APP PATH ACCOUNTS
-*
USE
NYACCTS ON FOCSU01
REGIONS ON FOCSU01
END

SET COMMIT = ON
-*
MNTCON PREPARESERVER
MNTCON LOADIMAGE NYACCTS
.
.
.
MNTCON STARTSERVER
```

Variable Varchar Data Type Support

WebFOCUS Maintain supports the variable varchar data type.

Variables are fully documented in *Using Variables in Procedures* in the *Developing Procedures* chapter in the *Developing WebFOCUS Maintain Applications* manual.

Additional Microsoft Access Support

In Release 5.2.3, WebFOCUS Maintain supports read/write access to Microsoft Access databases, date/time fields, and long text fields.

WebFOCUS Connector for Excel

In Release 5.2.3, the WebFOCUS Connector for Excel enables you to use Excel to interact with WebFOCUS Maintain and WebFOCUS Reporting. This product offering lets you load data to Excel using WebFOCUS, work with the data in Excel, and update the data centrally without having to leave Excel.

Use this product offering to develop and deploy Excel-based solutions that allow Excel users to share data in a common central database accessible to the enterprise. It is ideal for accounting, inventory, management, and performance applications—in fact, any application for which Excel is a primary tool.

The major benefits include:

- It allows Excel users to work in a familiar environment with no special training required.
- It eliminates manual cutting, copying, pasting, and rekeying of data.
- It simplifies automatic information reconciliation.
- It enables you to convert all current Excel-based solutions into enterprise-level applications that access any data source on any platform, without having to write code.

For instructions on implementing this product offering at your site, see your Information Builders representative.

CHAPTER 10

ETL Manager Enhancements

This chapter describes cumulative features for ETL Manager in Version 5 Release 2.

Topics:

ETL Server

- Request Generation
- Request Categories and Application Directories
- Request Logs
- Request and Log Management From the WebFOCUS Reporting Server Console
- Request Scheduling
- Bulk Loading
- Flat File Targets
- Migrating ETL Requests

ETL Workbench

- General Workbench Enhancements
- SQL/Answer Set Object
- Source Object
- Target Object
- Load Options
- Transport Simplification
- Nesting Dependent Requests
- E-mail Request Notification
- Optimized Load

Request Generation

ETL requests are generated at the time they are saved as stored procedures (FOCEXECs). This enhances the functionality of the ETL Engine in the following ways:

- Overhead required to run a request is greatly reduced. Smaller requests especially benefit from this with dramatically reduced run times.
- Requests can be run from the Interactive Agent Tool (edastart -t), which simplifies problem determination.
- You can edit requests to add functionality or to address problems.
- Saving and opening requests is much faster.
- Resource Analyzer can monitor requests by name.
- A FOCUS data source is no longer required for storing requests, which simplifies configuration. The log and statistics tables still use a FOCUS data source.
- Request names can be up to 32 characters, allowing for more descriptive names.
- A user ID can be up to 32 characters, increased from eight characters.

Request Categories and Application Directories

How to:

Create a New Application Directory

Set the Server Application Path

Previous releases of ETL Manager allow for the categorizing of requests at the time the request is saved. In this release, server application namespaces replace the categories. This affects:

- Directories where ETL requests are stored.
- Synonyms available to the ETL Manager user.

By default, ETL requests and synonyms are stored in the BASEAPP directory. To save requests in another directory, the directory must be:

- Created or mapped under APPROOT.
- Added to the application path in the server or user profile.

Application names are limited to 18 characters, must be alphanumeric or underscore, and must be lowercase only.

Requests migrated from Version 4.3.6 and Version 5.1 will have their categories converted to applications.

Procedure: How to Create a New Application Directory

1. In the Configure Application Path window, click *New Directory*.
2. Enter the name of the new directory, and click *OK*.

Procedure: How to Set the Server Application Path

1. In the APPROOT directory tree, drag and drop the directory you want to add from the list of directories to the APP PATH window.
2. Set the directory search order by moving the directories into their search order using the *Up*, *Down*, and *Remove* links.
3. Click *Set APP PATH*.

Request Logs

The request logs have been simplified and are easier to understand. Parameters for stored procedures appear in the log in addition to procedure names.

The logs capture output from user-written stored procedures (-TYPE commands).

Procedure: How to View the ETL Request Log From the ETL Workbench

1. Right-click your request in the ETL request folder.
2. Select *View Last Log* from the pop-up menu.

   The results are retrieved and displayed.

Request and Log Management From the WebFOCUS Reporting Server Console

You can manage ETL requests from the WebFOCUS Reporting Server Console. While you can still use the ETL Workbench to build and edit requests, you can perform most management activities from the WebFOCUS Reporting Server Console.

- **Request Management.** The WebFOCUS Reporting Server Console displays a list of all ETL requests stored on the server. You can select a request to view, run, or run deferred. You can also view all logs for a request.
- **Log Management.** You can view, create, or clean the log and statistics tables from the WebFOCUS Reporting Server Console. The Statistics report, which shows one line for each run of a request, is color-coded to indicate requests that are in progress (green), successfully completed (blue), or failed (red). Each report has a link to the detailed log for that run of the request. When viewing the statistics table, use the job ID to view the log for that job.

Procedure: How to Configure Request and Log Management

1. Click the *Configure ETL* link.

   The ETL Configuration window opens.
2. Make your changes for request management.
3. Make your changes for log management.

   The ETL Log and Statistics option specifies the action to take on the log and statistics tables. The options are:

   - **Create/Recreate** creates or recreates empty log and statistics tables. Create/Recreate is the default value.
   - **Clean** deletes all rows from the log and statistics tables up to a specified date.
   - **Backup** creates a backup of the log and statistics tables named ETLLOGBK and ETLSTABK, respectively.
   - **Deferred Statistics Recovery** moves information about deferred ETL requests to the log and statistics tables if the internal tables become corrupted.
4. Click *Submit*.
5. Click *Save and Restart*.

For complete details on the options in the ETL Configuration window, see *Configuring ETL Manager Behavior* in the *Using the iWay Web Console* chapter in the *iWay ETL Manager User's Guide*.

Request Scheduling

How to:

Schedule an ETL Request to Run Once

Schedule an ETL Request to Run on a Recurring Interval

Schedule an ETL Request to Run at Multi-Day Intervals

You can schedule the execution of your ETL requests with the ETL Manager Scheduler. Scheduled requests are initiated using the deferred receipt server, the same technology used by a majority of WebFOCUS installations.

The ETL Manager Scheduler supports schedule times to the minute and schedule intervals of minutes. The scheduler is controlled from the WebFOCUS Reporting Server Console ETL Configuration page.

- For scheduled requests to run, the Scheduler must be running. The etl_schedule_autostart radio button must be set to yes.
- The schedule interval determines how often the scheduler checks for requests to run. For example, for a request to run every five minutes, the scheduler interval etl_interval must be set to a value of 300 seconds or less.
- Changes made to the schedule interval on the WebFOCUS Reporting Server Console are not reflected on the schedule panel in the ETL Workbench until the program is shut down and restarted.

Procedure: How to Schedule an ETL Request to Run Once

1. In the Scheduler window, select *Run Once* in the Schedule Type section.
2. Select a start time in the Start Time section by entering the time in the fields, or using the arrow buttons.
3. Select a start date in the Start Time section by entering the date, or by clicking the calendar icon and choosing a date from the Calendar window.

Procedure: How to Schedule an ETL Request to Run on a Recurring Interval

1. In the Scheduler window, select *Recurring* in the Schedule Type section.
2. Select an interval type from the Interval Type section.
3. Select an interval number from the Interval Number section.
4. Select a start time in the Start Time section by entering the time in the fields, or using the arrow buttons.
5. Select a start date in the Start Time section by entering the date, or by clicking the calendar icon and choosing a date from the Calendar window.
6. Optionally, select an end time in the End Time section by entering the time in the fields, or using the arrow buttons.
7. Optionally, select an end date in the End Time section by entering the date, or by clicking the calendar icon and choosing a date from the Calendar window.

Note: You do not need to change the schedule to stop a query from executing regularly. Instead, set the status of the request to Inactive in the ETL request main window. For details on setting the status of an ETL request, see *Specifying ETL Request Properties* in the *Working With an ETL Request* chapter in the *iWay ETL Manager User's Guide*.

Procedure: How to Schedule an ETL Request to Run at Multi-Day Intervals

1. In the Scheduler window, select *Multi-Days* in the Schedule Type section.
2. Select the days of the week or month on which you want your job run from the Days of the Week or Days of the Month section.

   If you want your job to run at the end of the month, select *Last Day of Month*.
3. Select a start time in the Start Time section by entering the time in the fields, or using the arrow buttons.
4. Select a start date in the Start Time section by entering the date, or by clicking the calendar icon and choosing a date from the Calendar window that opens.
5. Optionally, select an end time in the End Time section by entering the time in the fields, or using the arrow buttons.
6. Optionally, select an end date in the End Time section by entering the date, or by clicking the calendar icon and choosing a date from the Calendar window that opens.

Note: You do not need to change the schedule to stop a query from executing regularly. Instead, set the status of the request to Inactive in the ETL request main window. For details on setting the status of an ETL request, see *Specifying ETL Request Properties* in the *Working With an ETL Request* chapter in the *iWay ETL Manager User's Guide*.

Bulk Loading

Bulk loading is supported for Sybase Adaptive Server IQ.

Bulk loading is fully documented in *Using Bulk Loading* in the *Building an ETL Request* chapter in the *iWay ETL Manager User's Guide*.

Flat File Targets

There is improved support for transformations to flat file targets. You can apply transformations in a single pass of the source data. Mapping to a predefined flat file target structure can be used.

Migrating ETL Requests

How to:

Migrate ETL Requests on UNIX and Windows

Migrate ETL Requests on OS/390

Migrate ETL Requests on OS/400

Enhancements have been made to the ETL migration features in the following areas:

- **Workbench.** As with previous releases, you can migrate requests from the ETL workbench. You can use copy-and-paste or drag-and-drop to migrate requests from one server to another.
- **Server.** A server procedure, initiated from the WebFOCUS Reporting Server Console, is provided to migrate requests from a Version 4.3 or Version 5.1 server to the new format in order to facilitate migration of a large number of requests.

You can migrate ETL Requests from version 4.x and 5.1.x when you upgrade ETL Manager to a newer version. The migration process includes the following tasks:

1. Migrate Master Files and Access Files.
2. Prepare the ETL internal database for migration.
3. Migrate the ETL Requests.
4. Test the migrated requests.

Although procedures vary slightly by operating system, each variation includes these tasks. For details, see *Migrate ETL Requests on UNIX and Windows* on page 10-8, *Migrate ETL Requests on OS/390* on page 10-10, *Migrate ETL Requests on OS/400* on page 10-12.

Note: Before you start the migration, review category names for the following special characters:

```
/\:*?"<>|
```

Categories names containing these special characters must be renamed before migration. In the migration, spaces in category names are changed to underscores (_).

Application names are limited to 18 characters.

Procedure: How to Migrate ETL Requests on UNIX and Windows

1. Locate the Master Files and Access files used by your ETL Requests. The default locations are:

Operating System and ETL Release	Default Location
UNIX, Release 4.3	home/ibi/srv43/cpm/catalog
UNIX, Release 5.1	home/ibi/srv51/etl/catalog
Windows, Release 4.3	c:\ibi\srv43\cpm\catalog
Windows, Release 5.1	c:\ibi\srv51\etl\catalog

2. Copy the Master Files and Access Files to the baseapp directory of APPROOT. This is the default directory for Release 5.2 ETL Requests.

   or

   Copy the Master Files and Access Files to any directory under APPROOT, then add that directory to APP PATH.

3. Prepare the ETL internal database for migration. If your ETL internal tables are stored in:

   A relational database (such as ORACLE, Microsoft SQL Server, or DB2). Configure a data adapter for the data source. For more information, see the documentation for the appropriate data adapter.

   FOCUS. No preparation is required if the internal database file (CMSCHED.FOC) and master file (CMSCHED.MAS) are in their original locations.

   Note that if you are migrating from a disconnected computer and have to copy these files, they should be copied to the *same* directory.

4. In the Web Console navigation pane, click *ETL*.

   The ETL Configuration page opens.

5. In the ETL Configuration page's navigation pane, click the *ETL Procedures* folder.

   A context menu opens.

6. Click *Migrate 4.x Procedures*.

   The Migrate page opens.

7. Enter the path to the directory where CMSCHED.MAS and either CMSCHED.ACX or CMSCHED.FOC files reside on the 4.*x* or 5.1.*x* Server. The default locations for each platform are as follows:

Operating System and ETL Release	Default Location
UNIX, Release 4.3	`home/ibi/srv43/cpm/copymgr`
UNIX, Release 5.1	`home/ibi/srv51/etl/etlmgr`
Windows, Release 4.3	`c:\ibi\srv43\cpm\copymgr`
Windows, Release 5.1	`c:\ibi\srv51\etl\etlmgr`

8. Click *Submit*.

   The message "Migrate procedure submitted" appears.

9. When the migration procedure finishes, review the log created to ensure that all ETL Requests migrated successfully. The following message should appear for each ETL Request:

   ```
   (ICM18164) Request NAME was created/updated successfully for user:
   USER.
   ```

   The following message should also appear:

   ```
   (ICM18354) Migration procedure completed successfully.
   ```

10. Review the list of new application directories. By default, ETL Requests are located in the APPROOT\baseapp directory. If a category name was previously assigned in the Properties window, a directory with that category's name is created under APPROOT and the following message appears:

    ```
    *-------   New Created APP directories  -------*/
    newapp
    *-----------------------------------------------*/
    ```

11. In the ETL Configuration page's navigation pane, click *Refresh Page*, then check to see that the application directories and requests are listed.

Procedure: How to Migrate ETL Requests on OS/390

1. Locate the Master Files and Access files used by your ETL Requests. These comprise all the data sets concatenated under the ddnames MASTER and ACCESS.
2. Allocate Master and Access PDSes using startup JCL at *qualif*.SRV52.ETL.DATA(EDASTART). For example:

   ```
   //MASTER DD DISP=SHR, DSN=qualif.MASTER.DATA
   //ACCESS DD DISP=SHR, DSN=qualif.ACCESS.DATA
   ```

   All the files from the concatenated datasets under ddnames MASTER and ACCESS will appear in the mapped application directory MVSAPP.
3. If the Master and Access files for the CMSCHED tables are in a PDS allocated in step 2, this step can be skipped.

   Optionally, copy the Master Files and Access Files for the CMSCHED tables from the PDS (for example, *qualif*.COPYMGR.MASTER.DATA) to an HFS directory (for example, /ibi/apps/baseapp/).

   In an OS/390 ISPF Command Shell session (option 6 from ISPF menu), enter the following command to copy the CMSCHED Master Files and Access Files:

   To copy the Master File:

   ```
   oput 'qualif.copymgr.master.data(cmsched)'
   '/ibi/apps/baseapp/cmsched.mas'
   ```

   To copy the Access File:

   ```
   oput 'qualif.copymgr.access.data(cmsched)'
   '/ibi/apps/baseapp/cmsched.acx'
   ```
4. Prepare the ETL internal database for migration. To do this, configure the Data Adapter for DB2 for the internal tables.
5. In the Web Console navigation pane, click *ETL*.

   The ETL Configuration page opens.
6. In the ETL Configuration page's navigation pane, click the *ETL Procedures* folder.

   A context menu opens.
7. Click *Migrate 4.x Procedures*.

   The Migrate page opens.

8. If you copied the Master Files and Access Files for the CMSCHED tables in Step 3, then enter the path to the directory where these files reside. The default location is:

   ```
   /ibi/apps/baseapp/
   ```

   If you did not copy the files in Step 3, and if the PDS was allocated in the start up JCL under MASTER and ACCESS ddnames, the default location is:

   ```
   'mas=//dd:master;acx=//dd:access'
   ```

9. Click *Submit*.

   The message "Migrate procedure submitted" appears.

10. When the migration procedure finishes, review the log created to ensure that all ETL Requests migrated successfully. The following message should appear for each ETL Request:

    ```
    (ICM18164) Request NAME was created/updated successfully for user:
    USER.
    ```

    The following message should also appear:

    ```
    (ICM18354) Migration procedure completed successfully.
    ```

11. Review the list of new application directories. By default, ETL Requests are located in the APPROOT\baseapp directory. If a category name was previously assigned in the Properties window, a directory with that category's name is created under APPROOT and the following message appears:

    ```
    *-------   New Created APP directories  -------*/
    newapp
    *-----------------------------------------------*/
    ```

    Note: To view or run ETL Requests in these new application directories they must be added to the server application path.

12. In the ETL Configuration page's navigation pane, click *Refresh Page*, then check to see that the application directories and requests are listed.

Procedure: How to Migrate ETL Requests on OS/400

1. Locate the Master Files and Access files used by your ETL Requests. The default location is /QSYS.LIB/EDAMAIN.LIB.
2. For each QSYS library, map the directory the new application will be located in into APPROOT using the Configure Application Path window, and then add the new directory to APP PATH.
3. Copy the Master Files and Access Files for the CMSCHED tables from the QSYS library (the default location for this library is /QSYS.LIB/EDAMAIN.LIB/MASTER.FILE) to the IFS directory. To do this, in an OS/400 session, enter the following command:

   To copy the Master file:

   ```
   CPYTOSTMF FROMMBR('/QSYS.LIB/EDAMAIN.LIB/MASTER.FILE/CMSCHED.MBR')
   TOSTMF('/home/ibi/srv52/etl/etlmgr/cmsched.mas')
   ```

   To copy the Access file:

   ```
   CPYTOSTMF FROMMBR('/QSYS.LIB/EDAMAIN.LIB/ACCESS.FILE/CMSCHED.MBR')
   TOSTMF('/home/ibi/srv52/etl/etlmgr/cmsched.acx')
   ```

   With the "replace" option:

   ```
   CPYTOSTMF FROMMBR('/QSYS.LIB/EDAMAIN.LIB/ACCESS.FILE/CMSCHED.MBR')
   TOSTMF('/home/ibi/srv52/etl/etlmgr/cmsched.acx') STFMOPT(*REPLACE)
   ```

4. After the Master Files and Access Files have been copied, edit CMCSHED.ACX and change the file name so that it points to the physical location of the table. This is done with the following:

   ```
   from FILENAME=CMSCHED
   to FILENAME=EDAMAIN/CMSCHED
   ```

   where:

   EDAMAIN

   Is the name of the library where the DB2 CMSCHED tables reside.

5. Prepare the ETL internal database for migration. To do this, configure the Data Adapter for DB2 for the internal tables.
6. In the Web Console navigation pane, click *ETL*.

   The ETL Configuration page opens.

7. In the ETL Configuration page's navigation pane, click the *ETL Procedures* folder.

   A context menu opens.

8. Click *Migrate 4.x Procedures*.

   The Migrate page opens.

9. Enter the path to the directory where CMSCHED.MAS and CMSCHED.ACX reside on the 4.*x* or 5.1.*x* Server. The default location is:

   ```
   /home/ibi/apps/baseapp
   ```

10. Click *Submit*.

    The message "Migrate procedure submitted" appears.

11. When the migration procedure finishes, review the log created to ensure that all ETL Requests migrated successfully. The following message should appear for each ETL Request:

    ```
    (ICM18164) Request NAME was created/updated successfully for user:
    USER.
    ```

    The following message should also appear:

    ```
    (ICM18354) Migration procedure completed successfully.
    ```

12. Review the list of new application directories. By default, ETL Requests are located in the APPROOT\baseapp directory. If a category name was previously assigned in the Properties window, a directory with that category's name is created under APPROOT and the following message appears:

    ```
    *-------   New Created APP directories  -------*/
    newapp
    *-----------------------------------------------*/
    ```

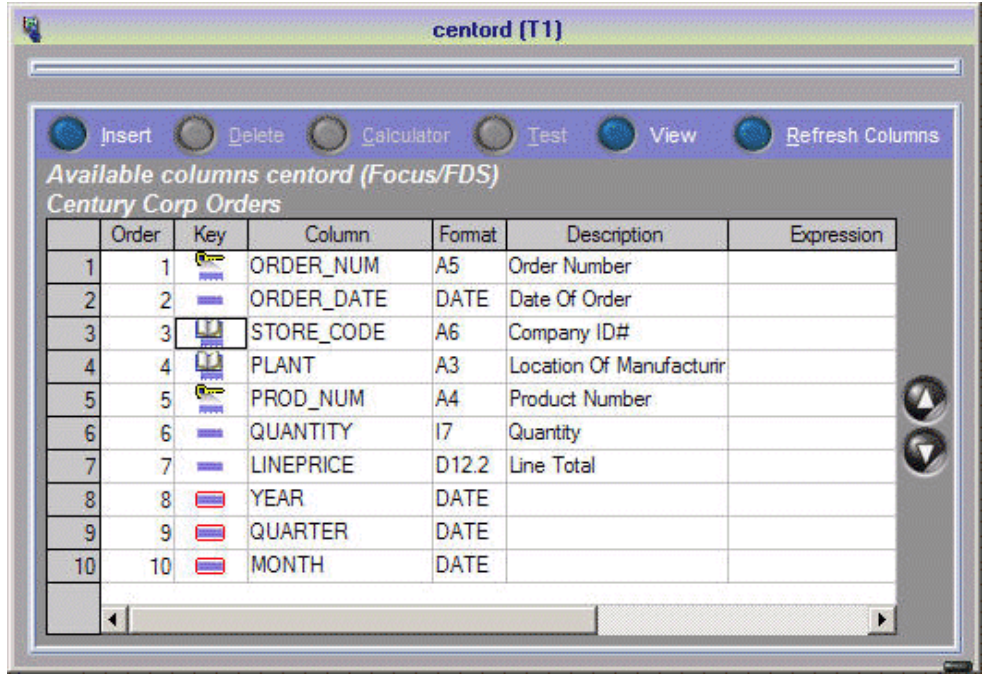
    Note: To view or run ETL Requests in these new application directories they must be added to the server application path.

13. In the ETL Configuration page's navigation pane, click *Refresh Page*, then check to see that the application directories and requests are listed.

General Workbench Enhancements

Enhancements to the Workbench include the following features:

- Column lists identify key columns not only for local tables but also for remote tables.
- You can use copy, cut, and paste operations in all grids to copy or move column names and transformations from one part of a request to another and between requests.
- Self-referential transforms (that is, CNTR = CNTR + 1) are no longer flagged as errors.
- You can use the TAB key to move between entry areas.
- Scroll bars are available in the workflow.
- Indexed columns in FOCUS files are identified.



SQL/Answer Set Object

The SQL/Answer object allows you to move SQL SELECT statements between applications by copying and pasting.

Source Object

You can use the LAST function in an extract transformation to compare the current and previous value of an input column. The LAST function retrieves the previous value for a field.

The Test button tests extract transformations.

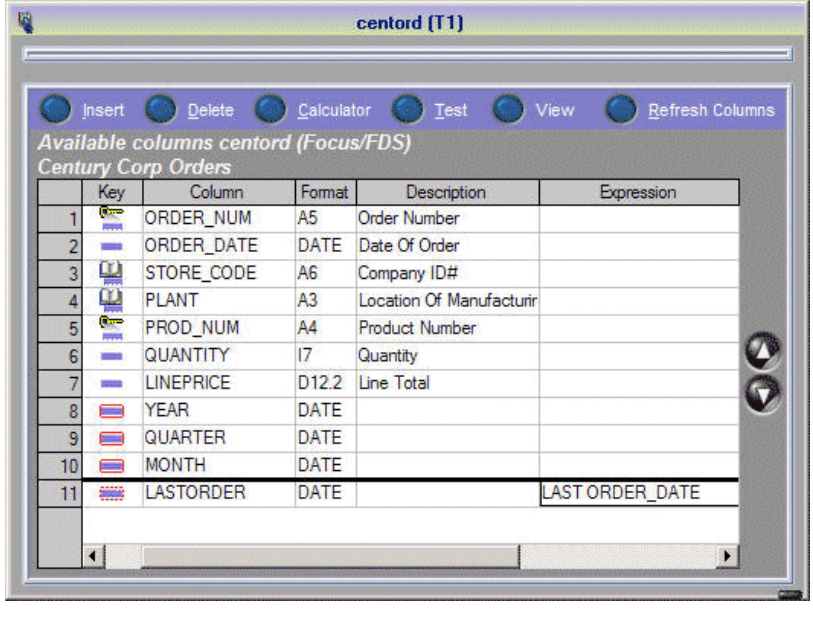
Procedure: How to Use the LAST Function in an Extract Transformation

1. In the Target window, click *Insert*.

   A new column appears in the Mapping and Transformation Rules grid.

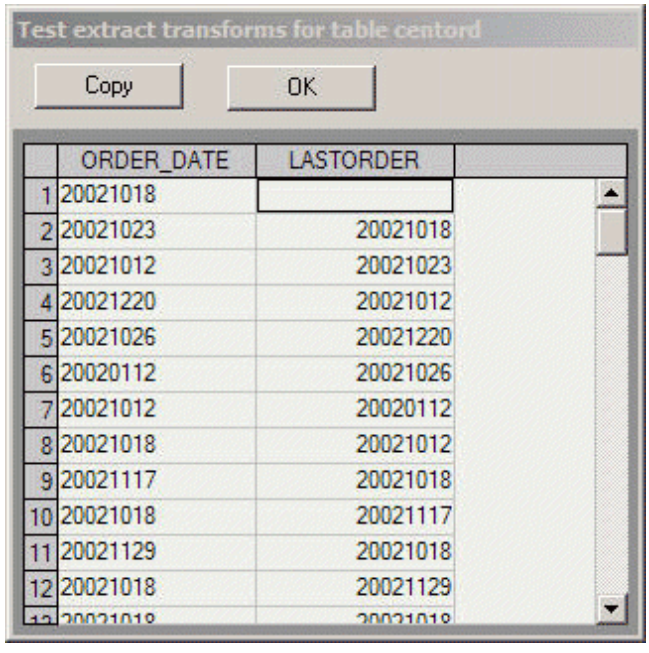
2. Double-click in the Column field and enter a name for the temporary column. In this example, the field name is LASTORDER.

3. Double-click the Format field and enter a format for the temporary column. In this example, the format is DATE.



4. In the Expression column, type LAST *fieldname*. In this example, LAST ORDER_DATE. You can also use the Transform Calculator to create this expression.

5. Select the new temporary field and click *Test*. The following window opens, showing the current and previous field values.



	ORDER_DATE	LASTORDER
1	20021018	
2	20021023	20021018
3	20021012	20021023
4	20021220	20021012
5	20021026	20021220
6	20020112	20021026
7	20021012	20020112
8	20021018	20021012
9	20021117	20021018
10	20021018	20021117
11	20021129	20021018
12	20021018	20021129

Target Object

For relational data source target tables, a pull-down menu provides a list of all connections, whether they are servers or data sources. The Validate option validates input rows as they are loaded into the target tables and sends rejected rows to a log file.

Validating Records

You can create a business rule that screens records as they are loaded into the data target. If a record does not meet the criteria, it is rejected.

When a value meets the criteria of the rule, a value of 1 is assigned to the record and the record is loaded. If the value does not meet the criteria of the rule, the value of 0 is assigned to the record and the record is rejected.

Procedure: How to Validate Records

1. In the Target (Mapping) window, click *Validate*.

   A column is added containing <VALIDATE> in the Column name field. An f is added to the first field of the next row. All expressions in this row and the rows beneath it will be rules that are applied to records as they are loaded.
2. Enter the column you want to apply the rule to in the Column Name field.
3. Enter the format in the Format field.
4. Enter the rule to apply to the specified field in the Expressions box, or click *Calculator* to create an expression with the calculator.
5. Optionally, enter a description in the Description field.

Load Options

How to:

Specify Load Options for a Relational Data Source

Load Options for a FOCUS or Fusion File

Prior to loading a relational table or a FOCUS or FUSION data source, you can specify the disposition of the target structure:

- For a relational table, the options are: Leave, Delete, and Truncate.
- For a FOCUS or FUSION data source, the options have been renamed to clarify their functions. They are Leave and Drop.

Access these options from the Target Setup tab in the Load component.

Procedure: How to Specify Load Options for a Relational Data Source

1. Add a Load component to the workspace, and connect it to the Target component.
2. Double-click the Load component to open it.

   The Load window opens with the Target Setup tab open.
3. If you want to activate Key Matching logic, select *Preprocessing*.

   The Key Matching Logic section appears.
4. In the Prior to Load section, select whether to remove data prior to loading the data target. The options are:

   - *Leave Table* does not delete the records already in a data target.
   - *Delete rows from table* deletes all rows prior to loading data. A DELETE FROM TABLE command is issued against the target, and a log of the deleted records is created so that a Rollback command can later be issued to recover them.
   - *Truncate* logically empties the table. A log is not created, so this is often faster than selecting *Delete rows from table*. However, it requires privileges that may not be in effect. If the Truncate command fails, the result is as if you had chosen to delete the records.

For complete details on the options in the Target Setup tab, see *Specifying Target Setup Options* in the *Building an ETL Request* chapter in the *iWay ETL Manager User's Guide*.

Procedure: How to Load Options for a FOCUS or Fusion File

1. Add a Load component to the workspace, and connect it to the Target component.
2. Double-click the Load component to open it.

   The Load window opens with the Target Setup tab open.
3. In the Prior to Load section, select whether to remove data prior to loading the data target. The options are:

   - *Leave Table* does not delete the records already in a data target.
   - *Drop Table* drops and recreates the data target. This option is available only for FOCUS and Fusion data targets.

For complete details on the options in the Target Setup tab, see *Specifying Target Setup Options* in the *Building an ETL Request* chapter in the *iWay ETL Manager User's Guide*.

Transport Simplification

An optional transport object simplifies specification of FTP parameters.

Procedure: How to Set Transfer Details for FTP

1. Add a Transport component to the workspace.
2. Connect the Transport component to the Column component, or to the Sort component, if you are using one.
3. Double-click the Transport component to open it.

   The Transport window opens.
4. In the Remote Directory field, enter the directory where the file to transfer is located.
5. In the Local Directory field, enter the directory where the file will be transferred.
6. In the User ID field, enter a valid user ID.
7. In the Password field, enter the password for the user ID.
8. Select *ASCII* or *Binary* from the Transfer Mode section.

Nesting Dependent Requests

Dependent requests can be nested. For example, request A can have request B as a dependency; request B can have request C as a dependency.

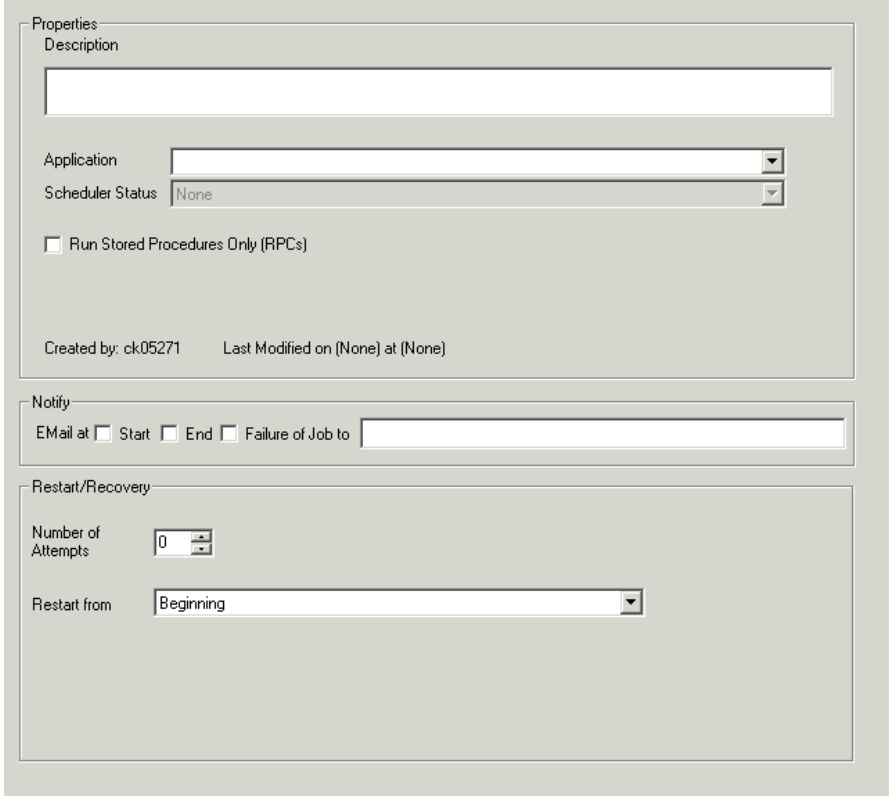
E-mail Request Notification

You can specify in the Workbench that an e-mail message be sent when a request starts, completes, or fails. An SMTP server is required to send e-mail notifications.

Set e-mail notification options from the Properties window.

Procedure: How to Setup E-mail Notification for an ETL Request

1. If the Properties window is not open, click the *General* button.
2. Select an option for e-mail notification in the Notify section. The options are:
   - *Start* sends an e-mail when an ETL request is executed.
   - *End* sends an e-mail when an ETL request's execution is successfully completed.
   - *Failure of Job to* sends an e-mail when a specified error occurs.



3. In the text box, enter the e-mail address of the person you want the e-mails sent to. Only one e-mail address can be entered and must be in the form *user@domain*.com.

For complete details on the options in the Properties window, see *Setting Properties of an ETL Request* in the *Building an ETL Request* chapter in the *iWay ETL Manager User's Guide*.

Optimized Load

How to:

Specify Optimized Load for a Relational Data Source

Loading relational data source targets is much faster when optimized load capabilities are used to insert a block of rows at once. For Microsoft SQL Server and Sybase, Fastload is used; for other relational data sources, Array Insert is used.

Optimized load is available for relational data source targets on Windows and UNIX. To utilize optimized load for these platforms, be sure that *Include duplicates* is selected on the load object so that no pre-processing is performed.

A user entry area labeled *Optimize Load* accepts the number of rows to load at once as input.

- If the number one (1) is entered (the default), row-at-a-time insert is used and the behavior is the same as in previous releases.
- If a number greater than one (1) is entered, optimized load is used.

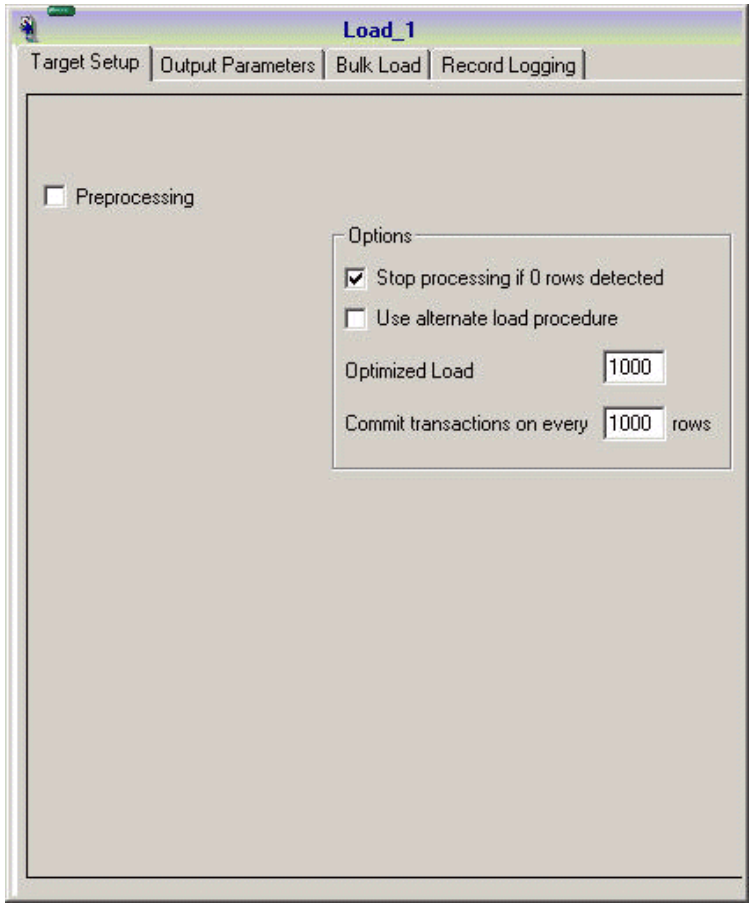
Note that when optimized load is used:

- Loading is much faster since a block of rows are inserted at once.
- Input data should be clean. If any one row in the block causes a data source constraint violation such as not null or unique index, the entire block is rejected.
- No row counts are available for the number of records inserted or rejected in the detail log or statistics. N/A (not available) will appear in the log instead.
- Record logging (used to write rejected records to a flat file for review) is not available.

Procedure: How to Specify Optimized Load for a Relational Data Source

1. Add a Load component to the workspace, and connect it to the Target component.
2. Double-click the Load component to open it.

   The Load window opens with the Target Setup tab open.
3. In the Options area, select *Optimized load* and enter:
   - *1* inserts records a row at a time. 1 is the default value.
   - A number greater than one uses optimized load.



For complete details on the options in the Target Setup tab, see *Specifying Target Setup Options* in the *Building an ETL Request* chapter in the *iWay ETL Manager User's Guide*.

CHAPTER 11

Resource Analyzer and Resource Governor Enhancements

This chapter describes cumulative features for Resource Analyzer and Resource Governor in Version 5 Release 2.

Topics:

- Configuring Internal Tables From the WebFOCUS Reporting Server Console
- Accessing Resource Analyzer and Resource Governor Reports From the WebFOCUS Reporting Server Console
- Increased Support for Long Procedure Names
- Resource Analyzer Reports

Configuring Internal Tables From the WebFOCUS Reporting Server Console

In this section:

Granting Access to Resource Analyzer/Resource Governor Data Sources

Running the Resource Governor Configuration Verification Program (CVP)

How to:

Create Resource Analyzer/Resource Governor Internal Tables

Grant Access to an Administrative Data Source

Run the Resource Governor Configuration Verification Program (CVP)

Server administrators can configure the Resource Analyzer and Resource Governor internal tables from the WebFOCUS Reporting Server Console. The configuration process prompts you for the information required to create these tables. You can turn Global Request Monitoring and Global Stored Procedure (FOCEXEC) monitoring on or off when the internal tables are created.

Procedure: How to Create Resource Analyzer/Resource Governor Internal Tables

1. Do one of the following:

   - From the WebFOCUS Reporting Server Console, select *Workspace* followed by *Resource Analyzer/Governor*.

   or

   - Go to the Resource Analyzer/Resource Governor Administrator and connect to the appropriate server.

2. Enter the information that is required to create the Resource Analyzer and Resource Governor internal tables in the available fields. The internal tables are created based on your entries. For more information on any of these fields, click the ? icon. A message appears when this process is complete.

Next Steps:

- If you are storing your internal tables in a relational database management system, go to *Granting Access to Resource Analyzer/Resource Governor Data Sources* on page 11-3.
- If you selected Resource Governor during installation, go to *Running the Resource Governor Configuration Verification Program (CVP)* on page 11-4 before you use the Resource Governor Administrator to control monitoring and build rules.
- If you are configuring for Resource Analyzer only, you do not need to run the CVP after the internal tables are created. You may use the Resource Analyzer Administrator to control the monitoring of data sources.

Granting Access to Resource Analyzer/Resource Governor Data Sources

Resource Analyzer/Resource Governor has two separate data sources; one used for administration and the other for usage monitoring data. If you are using a relational data source (that is, not FOCUS) as your Resource Analyzer/Resource Governor repository, a GRANT command is issued for all users (PUBLIC) when the internal tables are created. This command enables all users to write to the usage-monitoring data source.

The owner ID specified when the internal tables are created is granted user-access rights to the administrative data sources (SMCONTROL, SMKBASE, SMPRL, and SMPARAMETERS).

If you want to grant other users access to the Resource Analyzer/Resource Governor administrative data sources, you must issue additional GRANT commands for those user IDs. A sample data file, gkegrant.dat, can be found in the /etc directory in EDAHOME. This file contains the commands needed to grant user access to the Resource Analyzer and Resource Governor administrative data sources.

Syntax: How to Grant Access to an Administrative Data Source

```
GRANT SELECT, UPDATE, INSERT ON owner.admin_datasource TO user1, user2 ..
```

where:

owner

Is the owner name of tables that are used for system administration and collection.

admin_datasource

Is one of the Resource Analyzer/Resource Governor administrative data sources (SMCONTROL, SMKBASE, SMPRL, and SMPARAMETERS).

user1, user2 ..

Are the user IDs of the users to whom you wish to grant access.

Note: To grant certain users access to the Resource Analyzer and Resource Governor administrative data sources, you must issue the GRANT command for each data source.

Running the Resource Governor Configuration Verification Program (CVP)

Note: The CVP is applicable *only* to Resource Governor, *not* Resource Analyzer.

The CVP is a program that verifies the Resource Governor installation and configuration. The following remote procedures, located in the catalog directory of EDAHOME, are executed:

- gkecol
- gkeparm
- gkegov
- gkerule
- gkedel

The CVP verifies the following:

- Request usage monitoring (collection).
- Request governing based on collection.

The CVP uses a temporary table, GKEIVP, for validating usage monitoring and governing.

Procedure: How to Run the Resource Governor Configuration Verification Program (CVP)

1. Run the procedure, gkeivp.fex, located in the catalog subdirectory of the EDAHOME directory. (In the WebFOCUS Reporting Server Console, you can add this directory to the server path, and then click the procedure in the Procedures page and select *Run*.)
2. Examine the output. If the Resource Governor installation and configuration is successful, the following messages are included among the output:

   - This statement indicates that the test data has been put into the test data source:

     ```
     ***********************************
     * INSERTS COMPLETED FOR TEST DATA *
     ***********************************
     ```

   - This statement indicates that the Resource Governor utility procedure, gkecol, has successfully completed, and that data about test requests will be temporarily logged in the Usage Monitor:

     ```
     *********************************
     * GKECOL COMPLETED SUCCESSFULLY *
     *********************************
     ```

   - This statement indicates that the test requests, using SELECT against the GKEIVP test data source, have completed successfully, and that Usage Monitor data has been populated in the Resource Governor Usage Monitor data sources:

     ```
     *********************************************
     * REQUESTS COMPLETED FOR TEST COLLECTION DATA *
     *********************************************
     ```

   - This is a cancellation message that Resource Governor generates to indicate that a SELECT statement was issued and canceled (based on rules built by the CVP and used for governing):

     ```
     (GKE36048) RESOURCE GOVERNOR CANCELED THIS REQUEST. Governing Mode
     = GOVERN
     KBName = IVP RuleNum = 3 Threshold Type = ROWS Thresh Exceeded = 10
     ```

If the Resource Governor installation and configuration is unsuccessful, messages (preceded by the keyword ERROR) appear along with the output. These messages are specific to the type of error encountered. For example:

- The following message results from the incorrect execution of the gkecol procedure:

  ```
  ERROR SETTING COLLECTION ON
  ```

- The following message results from the incorrect execution of the GKETABLE procedure:

  ```
  ERROR CREATING GKETABLE
  ```

Tip: If the success messages are not found in the TS3OUT listing, or if error messages are found, turn on tracing and examine the messages in the trace for any problems in the setup.

Accessing Resource Analyzer and Resource Governor Reports From the WebFOCUS Reporting Server Console

Resource Analyzer and Resource Governor reports are accessible from the WebFOCUS Reporting Server Console using the HTTP Listener.

For complete details about accessing Resource Analyzer and Resource Governor reports, see the *iWay Resource Analyzer Administrator's and User's Manual* and the *iWay Resource Governor Administrator's and User's Manual*.

Increased Support for Long Procedure Names

Resource Analyzer and Resource Governor monitor long procedure names for WebFOCUS applications. In addition, Resource Analyzer and Resource Governor store up to 66 bytes of a stored procedure name and all reports support this length.

Resource Analyzer Reports

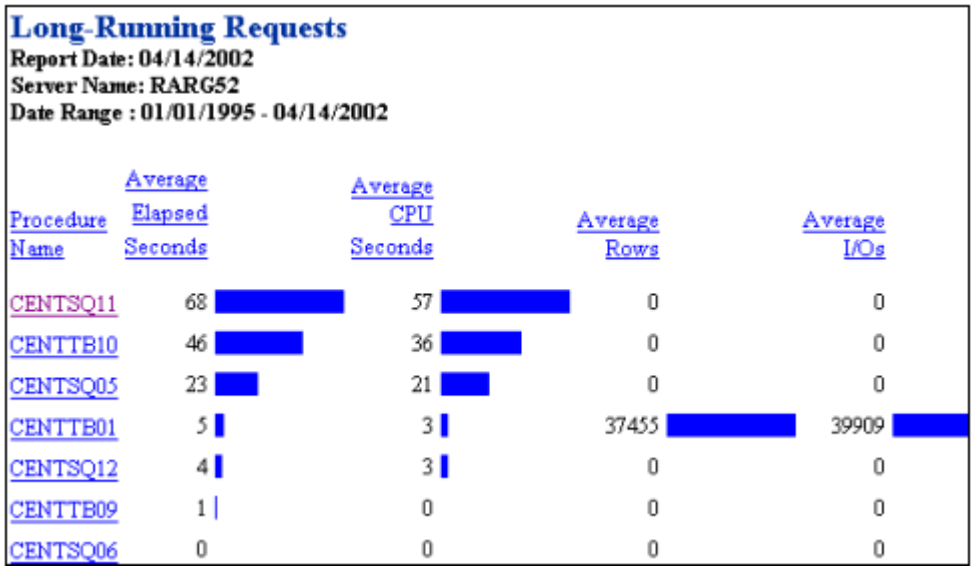
The following reports have been added to Resource Analyzer:

- **Long-running Requests** uncover procedures which use excessive resources that result in long waits for users.
- **Large Volume Requests** identify requests returning more data than necessary and target them for optimization.
- **Candidates for Column Index** identify columns that should be indexed.

Long-running Requests Report

Procedures with long running requests use excessive resources, a particular problem when dealing with operational systems. These procedures should be targeted for optimization.

When you click the *Long-running Requests* hyperlink, the Long-running Request report opens:



Long-Running Requests
Report Date: 04/14/2002
Server Name: RARG52
Date Range : 01/01/1995 - 04/14/2002

Procedure Name	Average Elapsed Seconds	Average CPU Seconds	Average Rows	Average I/Os
CENTSQ11	68	57	0	0
CENTTB10	46	36	0	0
CENTSQ05	23	21	0	0
CENTTB01	5	3	37455	39909
CENTSQ12	4	3	0	0
CENTTB09	1	0	0	0
CENTSQ06	0	0	0	0

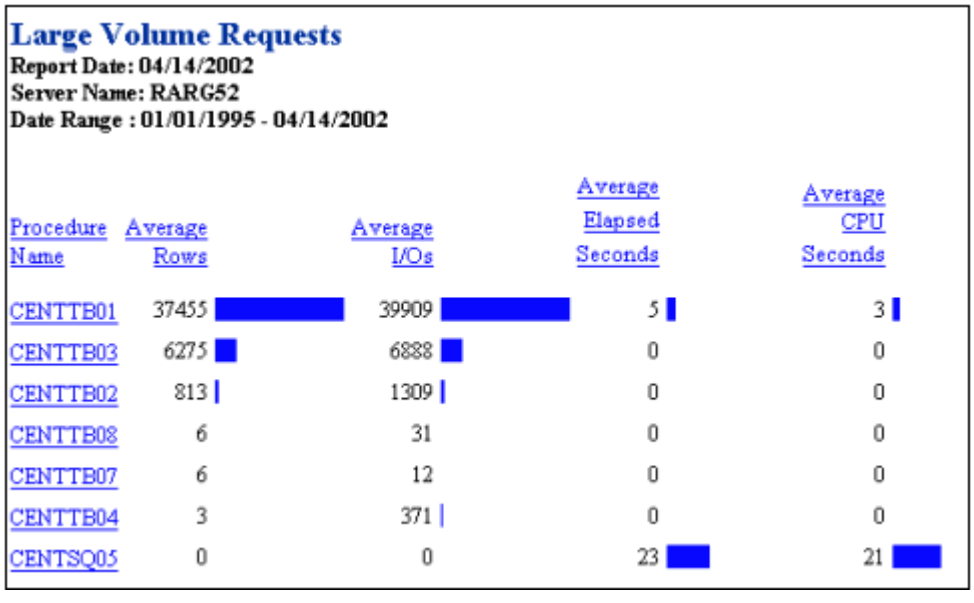
By default, the report is sorted by Average Elapsed Seconds and Average CPU Seconds.

In this report, the procedure names are hyperlinks that allow you to drill down to a report that shows the requests that were executed from the selected procedure. From this report, you can drill down further to see the actual request syntax.

Large Volume Request Reports

Large volume requests are requests that return more data than might be necessary. These requests can be targeted for optimization. You can identify these reports with the Large Volume Requests Report.

When you click the *Large Volume Requests Report* hyperlink, the Large Volume Requests report opens:



Large Volume Requests

Report Date: 04/14/2002
Server Name: RARG52
Date Range : 01/01/1995 - 04/14/2002

Procedure Name	Average Rows	Average I/Os	Average Elapsed Seconds	Average CPU Seconds
CENTTB01	37455	39909	5	3
CENTTB03	6275	6888	0	0
CENTTB02	813	1309	0	0
CENTTB08	6	31	0	0
CENTTB07	6	12	0	0
CENTTB04	3	371	0	0
CENTSQ05	0	0	23	21

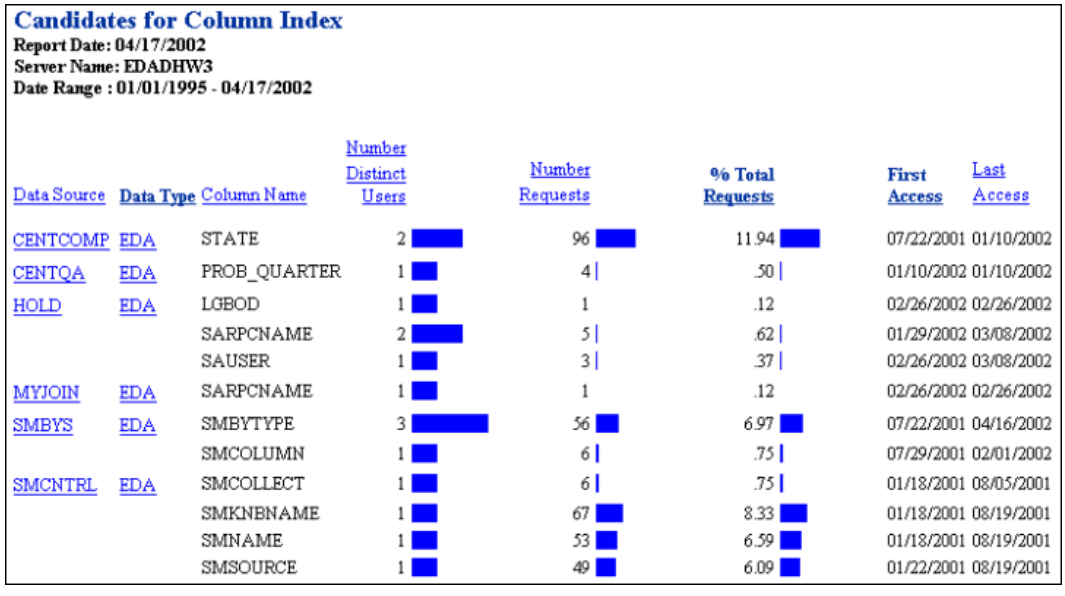
By default, the report is sorted by Highest Average Rows.

In this report, the procedure names are hyperlinks that allow you to drill down to a report that shows the requests that were executed from the selected procedure. From this report, you can drill down further to see the actual request syntax.

Candidates for Column Index Report

Indexing columns that are frequently used in selection criteria for reports can improve efficiency. This report identifies columns that should be indexed.

When you click the *Candidates for Column Index* hyperlink, the Candidates for Column Index report opens:



Candidates for Column Index

Report Date: 04/17/2002
Server Name: EDADHW3
Date Range : 01/01/1995 - 04/17/2002

Data Source	Data Type	Column Name	Number Distinct Users	Number Requests	% Total Requests	First Access	Last Access
CENTCOMP	EDA	STATE	2	96	11.94	07/22/2001	01/10/2002
CENTQA	EDA	PROB_QUARTER	1	4	.50	01/10/2002	01/10/2002
HOLD	EDA	LGBOD	1	1	.12	02/26/2002	02/26/2002
		SARPCNAME	2	5	.62	01/29/2002	03/08/2002
		SAUSER	1	3	.37	02/26/2002	03/08/2002
MYJOIN	EDA	SARPCNAME	1	1	.12	02/26/2002	02/26/2002
SMBYS	EDA	SMBYTYPE	3	56	6.97	07/22/2001	04/16/2002
		SMCOLUMN	1	6	.75	07/29/2001	02/01/2002
SMCNTRL	EDA	SMCOLLECT	1	6	.75	01/18/2001	08/05/2001
		SMKNBNAME	1	67	8.33	01/18/2001	08/19/2001
		SMNAME	1	53	6.59	01/18/2001	08/19/2001
		SMSOURCE	1	49	6.09	01/22/2001	08/19/2001

This report shows how often columns were used as selection criteria in requests.

Click any of the hyperlinked column headings to resort the report by that column.

In this report, the data source names are hyperlinks that allow you to drill down to a report that shows usage for the columns for the specified data source, including whether they were used with functions, relations, selects, sorts, groups, or never used. From this report, you can drill down further to see more information about when the columns were used.

APPENDIX A

New Feature Roadmap for WebFOCUS Version 5 Release 2.3

The *Roadmap* appendix is intended as a chapter-by-chapter reference for users who wish to quickly locate information about features introduced in Release 5.2.3.

Topics:

- WebFOCUS Reporting Language Enhancements
- Developer Studio Enhancements
- Managed Reporting and Dashboard Enhancements
- Ad Hoc Reporting Enhancements
- Report Caster Enhancements
- WebFOCUS Client Enhancements
- WebFOCUS Reporting Server Enhancements
- Data Adapter Enhancements
- WebFOCUS Maintain Enhancements

WebFOCUS Reporting Language Enhancements

Enhancement:	See:
Multivariate regression	*Calculating Trends and Predicting Values* on page 1-2.
IN-RANGES-OF	*Grouping Sort Field Values in Ranges* on page 1-27.
TABLASTPAGE variable	*Indicating Total Number of Pages* on page 1-29.
Summarizing alphanumerics	*Manipulating Summary Values* on page 1-31.
SET HNODATA	*Controlling Missing Values Propagated to a HOLD File* on page 1-43.
SET NULL	*Propagating Missing Values to a Variable Length Delimited HOLD File* on page 1-48.
AnV fields	*Using Variable-Length Character Expressions* on page 1-50.
SET USERFNS SET USERFCHK	*Validating Function Arguments* on page 1-66.
FML row indents	*Indent Row Titles in an FML Hierarchy* on page 1-86.
Headings and footings	*Increased Numbers of Headings and Footings* on page 1-98.
ACROSS columns	*Increased Number of Columns Generated by ACROSS* on page 1-98.
Column titles	*Increased Space for Column Titles* on page 1-98.
DEFINE compiler	*Compiling Expressions Containing DEFINEs* on page 1-105.
SET FOCHTMLURL	*Establishing a Non-Standard Alias* on page 1-105.
SET FILECASE	*Converting Case for File Names and Directories* on page 1-107.
Multiple Drill-down links	*Specifying Multiple Drill-Down Links* on page 1-140.
Minus Edit display format	*Displaying a Minus Sign After Negative Data* on page 1-124.
FORMAT TAB	*TAB Format* on page 1-153.

Developer Studio Enhancements

Enhancement:	See:
Managed Reporting development	*Managed Reporting Developer for Windows* on page 2-3.
Output formats	*Comma-delimited and Tab-delimited Output Formats* on page 2-12.
Storing data and components in ANSI (Windows) format	*Configuration for ANSI (Windows) Format* on page 2-13.
Apportioning numeric data in ranges	*Summary of Report Painter New Features* on page 2-14.
Multiple drill down links	*Creating Multiple Drill Downs With the Report Painter* on page 2-21.
Chaining parameters for record selections	*Filtering Dynamic Parameter Values* on page 2-40.
Image hyperlinks in Resource Layout Painter (RLP)	*Adding an Image With a Hyperlink to the Layout* on page 2-43.
Multi-column display of radio buttons and check boxes in RLP	*Displaying Radio Buttons and Check Box Lists in Multiple Columns* on page 2-44.
Dynamic display of names in RLP	*Specifying Different Names for Dynamic Parameter Values* on page 2-46.
Hiding borders around frames in RLP	*Hiding Borders Around Frames* on page 2-47.
Using parameter values with external reports in RLP	*Applying Parameter Values to External Reports* on page 2-47.
Update Assist enhancements	*Validation of User-Supplied Data in Update Assist Applications* on page 2-51. *Choosing a Color Scheme for an Update Assist Application* on page 2-52. *Customizing Update Assist Designs* on page 2-52. *Calendar Control for Date-Formatted Fields in Update Assist* on page 2-53. *Right-Click Menu Options on Trees* on page 2-53. *Sorting Tables* on page 2-54.
Increased number of columns generated by Across	*Summary of Report Painter New Features* on page 2-14.

Managed Reporting and Dashboard Enhancements

Enhancement:	See:
Developer user type (previously Domain Admin)	*Developer User Type* on page 3-2.
Windows-based Managed Reporting development	*Managed Reporting Developer for Windows* on page 3-3.
Moving application components between environments	*Change Management* on page 3-4.
Trusted connections between Managed Reporting and ReportCaster	*Servlet Exit* on page 3-5.
Configuring authentication of users against passwords defined in an LDAP directory	*LDAP Authentication* on page 3-6.
Signon integration	*Integrating Managed Reporting Security With Basic Web Server Authentication* on page 3-5.
Custom Reports	*Custom Reports in Dashboard* on page 3-19.
-INCLUDE in Custom Reports	*Execution of a Custom Report Using -INCLUDE* on page 3-11.
Filters for Standard Reports	*Filters For Standard Reports in Dashboard* on page 3-20.
Content pages for viewing ReportCaster and Report Library interfaces	*Content Pages: ReportCaster and Report Library* on page 3-20.
Dynamic tracing	*Dynamic Tracing* on page 3-20.

Ad Hoc Reporting Enhancements

Enhancement:	See:
HTML Report Assistant	*End User Drill Down Capability* on page 4-16. *Performing OLAP Operations Directly From the Report* on page 4-16.
HTML Graph Assistant	*Addition of Several Templates to the Styles Tab* on page 4-18. *Updated Styles Tab to Include Graph Types and Graph Styles* on page 4-18. *Addition of the Preview Screen* on page 4-18. *Auto Update Feature Added to the Preview Screen* on page 4-19. *Creating a WHERE Statement Using Date and Time Values* on page 4-20.

Report Caster Enhancements

Enhancement:	See:
Repository migration	*Migration Tools* on page 5-2.
Configuring the Web Application	*Configuring the ReportCaster Web Application* on page 5-2.
Configuration menu options	*Configuration Tools* on page 5-3.
Signon integration	*Security Configurations* on page 5-3.
Security Exit integration	*ReportCaster Authentication Exit* on page 5-4.
Choosing a ReportCaster user interface	*Configuring Access to the User Interfaces* on page 5-4.
Scheduling	*Distribution Enhancements When Creating a Schedule* on page 5-7. *Scheduling Wizard and HTML User Interface* on page 5-8. *Distribution Enhancements When Using the Scheduling Wizard* on page 5-8.
Changing execution IDs	*ReportCaster Console* on page 5-10.
ReportCaster API	*ReportCaster API* on page 5-17.

WebFOCUS Client Enhancements

Enhancement:	See:
Server-side graphics	*Configuring Special Services* on page 6-9.
Auto-prompting for variables	*Configuring Special Services* on page 6-9.
Security and node behavior settings in odin.cfg	*Add a New Node to ODIN.CFG* on page 6-4.
Sorting the Server list	*Change a Node* on page 6-6.
Positional flags (MR_PGM_FLAGS) for custom security exits	*Configuring WebFOCUS Startup Parameters* on page 6-10.
Settings for alias, NLS characters, signon prompting, and encryption	*Changing WebFOCUS Client Settings* on page 6-15. *Redirection Settings* on page 6-18. *Customizing WebFOCUS Client Settings* on page 6-20. *NLS Settings* on page 6-20.
Double-byte character set (DBCS) code pages	*National Language Support for International Computing* on page 6-23.
Translated interfaces: Japanese, Korean, simplified and traditional Chinese	*National Language Support for International Computing* on page 6-23.
Language switching	*Using the Dynamic Language Switch* on page 6-24.

WebFOCUS Reporting Server Enhancements

Enhancement:	See:
Remote installation	*Installing WebFOCUS Reporting Server Remotely* on page 7-2.
Remote control	*Controlling WebFOCUS Reporting Server Remotely* on page 7-2.
Log file creation and archiving	*Specifying the Number of Lines in the EDAPRINT Log* on page 7-16.
Listener recovery from an operating server	*Automatic Self Recovery* on page 7-17.
Lawson security rules	*Lawson Security Adapter* on page 7-25.
Security Integration Architecture on Tru64 operating systems	*C2 Security Support* on page 7-25.

Data Adapter Enhancements

Enhancement:	See:
Adabas	*Setting Connection Attributes* on page 8-2. *Adabas Hyper and Phonetic Descriptors* on page 8-5. *Switching Access Mode Using NOPACCESS* on page 8-8.
DB2	*Declaring DB2/CAF Connection Attributes From the WebFOCUS Reporting Server Console* on page 8-11. *Support for DB2 8.1* on page 8-13.
Essbase	*Preventing Aggregation of Non-Consolidating Members* on page 8-16. *Suppressing Zero Values* on page 8-17. *Suppressing Missing Data* on page 8-17. *Suppressing Zero Values and Missing Data* on page 8-18.
J. D. Edwards	*Data Adapter for J.D. Edwards* on page 8-19.
Millennium	*Data Adapter for Millennium* on page 8-19.
PeopleSoft	*Data Adapter for PeopleSoft* on page 8-20.
SAP R/3	*Aggregation Support in the SAP R/3 Interface* on page 8-23.
VSAM	*Data Adapter for VSAM* on page 8-25.

Enhancement:	See:
Java classes	*Java Applications Adapter* on page 8-26.
Database selection	*Database Selection From the Console* on page 8-33.
SUBSTRING function	*SUBSTRING Function Behavior* on page 8-42.

WebFOCUS Maintain Enhancements

Enhancement:	See:
Smart deployment	*Deploying Only Changed Files With Smart Deploy* on page 9-4.
Parsing activity	*Customizing Parsing Activity* on page 9-8.
Text search	*Searching for Text in a Project* on page 9-9.
Cascading Style Sheets in forms	*Applying a Cascading Style Sheet to a Form* on page 9-10.
Testing procedures	*Testing Procedures With Run Procedure* on page 9-37.
Search paths	*Specifying Additional Search Paths* on page 9-38.
Microsoft Access integration	*Additional Microsoft Access Support* on page 9-43.
Excel integration	*WebFOCUS Connector for Excel* on page 9-44.

Index

Symbols

.fcm extension in Maintain 9-36

.for files in Maintain 9-12

.gfa files 9-4
    deployment information 9-4

.htm files 6-33

.mnt extension in Maintain 9-36

A

accessing data sources concurrently in Maintain 9-42

accessing Developer Launch Console in Maintain 9-7

accessing Java classes 8-26

ACROSS field values 1-149

ACROSS phrase 1-98, 2-20

ACROSS-TOTAL phrase 1-149 to 1-150

active connections 7-4

Adabas Data Adapter 8-2
    creating synonyms manually 8-8
    format buffers and 8-7
    GFBID (Global Format Buffer ID) 8-8
    Global Format Buffer ID (GFBID) 8-8
    Internal Sequence Number (ISN) 8-6, 8-8
    inverted lists 8-5
    ISN (Internal Sequence Number) 8-6, 8-8
    OCCURS attribute for Adabas 8-6
    ORDER field 8-6
    setting new synonyms 8-4
    superdescriptors and 8-4

ADD command 1-78

Add to Favorites option 3-15

adding objects to HTML pages 2-43
    frames 2-47
    images 2-43

Advanced privilege 3-8

agent services 7-2

aligning decimal points 1-111

Allbase Data Adapter 8-9

alpha fields 8-38
    restricting size 8-38

alphanumeric data 1-50
    and functions 1-52 to 1-54, 1-61

altering X and Y axis values 1-97

alternate indexes 8-25
    selecting records and 8-25

alternate server nodes 6-8

amper auto-prompting 1-99

amper variables 1-99

ANSI format 2-13

AnV fields 1-52
    and functions 1-52, 1-54, 1-56, 1-59, 1-61, 1-63

application directories 10-2
    creating 10-3
    name spaces 10-2

Application Path 3-7

application-level tracing in Maintain 9-16

applications 2-49
    accessing concurrently in Maintain 9-42
    tracing in Maintain 9-16
    updating with Update Assist tool 2-49

applications and projects 9-2

applying Cascading Style Sheets to forms 9-11

applying classes to controls 9-11

applying time formats 2-9

architecture of Developer Studio 2-2

authentication 7-21

auto-increment columns in Maintain 9-40

automatic drill down 4-5

B

Baltic language support 7-7, 7-12 to 7-13
    configuring 7-15

banners 3-14

BAPIs (Business Application Programming Interfaces) 8-22

binding variables in Maintain 9-17

BLANKINDENT SET parameter 1-85, 2-69

BORDER attribute 1-107 to 1-108, 1-110

borders 1-107, 1-110, 2-47, 2-65, 2-67 to 2-68
    around headings and footings 1-108
    colors 1-108
    hiding 2-47
    styles 1-108
    width of 1-108

bulk load capacity 10-7

Business Application Programming Interfaces (BAPIs) 8-22

BW Reporting 8-21

C

C2 server security 7-25

CA certificates 7-24

caching parameters 2-42

calculated values 1-38
    and prefix operators 1-38

calculating trends 1-2
    and multivariate regression 1-18
    with FORECAST 1-3

calculating trends with FORECAST 2-26

calendar in Update Assist 2-53

CALL statements in Maintain 9-38

CALLJAVA command 8-26
    execute method 8-27
    fetch method 8-27

Candidates for Column Index report 11-7, 11-9

CAPTION parameter 2-56

captions 2-57
    indenting 2-56

captions in a hierarchy 2-56 to 2-57

Cascading Style Sheets (CSS) in Maintain 9-10

CDN (Continental Decimal Notation) 1-118 to 1-119
    punctuating large numbers 1-118

CDN parameter 2-69

CDN parameter in Maintain procedure 9-20

cells 1-89
    adding borders 2-68
    formatting 1-89 to 1-90, 2-64, 2-67 to 2-68
    identifying 1-89, 2-64

CENT-ZERO parameter 1-119, 2-69

CGI aliases in Maintain 9-3

CGI parameters in Maintain 9-16

CGI scripting in Maintain 9-3

CGI/Servlet 1-99
    prompting for variables 1-99

chain feature 2-40 to 2-42
    caching parameters 2-42
    removing parameters from a chain 2-41
    reversing the order of parameters 2-41

Change Management 3-4

changing date formats 2-8

changing settings from Maintain procedure 9-21

changing time display 2-9

CHAR2INT Maintain function 9-33

character expressions 1-50
    variable length 1-50

character functions 1-155
    SQUEEZ 1-155 to 1-156
    STRIP 1-157
    TRIM 1-158 to 1-159

character strings 1-155
    reducing spaces 1-155

Check Box Properties window 2-44

check boxes 2-44
    Check Box Properties window 2-44

CLOB fields 8-42

COBOL Copy Book 8-32

COBOL FD Translator 8-32

code page generation files 7-5 to 7-6

code page values 7-5 to 7-6

code page values for OS/400 7-13

code page values for UNIX 7-12

code page values for Windows 7-12

code pages 7-5 to 7-6
    selecting 7-6
    specifying manually 7-10
    supported 7-7

color schemes for Update Assist applications 2-52

column titles 1-98

column totals 1-149 to 1-150
    EXL2K format 1-133

columns 1-89
    formatting 1-89, 1-138, 2-64 to 2-65, 2-67
    identifying 1-89, 2-64 to 2-65

COM data source 1-151, 2-12

COM format 1-152

combining mixed format reports 1-125 to 1-126

combining multiple reports 1-125, 2-11

combining PDF reports 1-125

combining report formats 1-126

Comma Suppress Edit Format Option 2-9

comma-delimited data sources 2-12

commands 4-10
    COMPUTE 4-10
    DEFINE 4-10

COMMIT parameter in Maintain procedure 9-20

compiling a Java program 8-27

compiling expressions 1-105

compound reports 1-125, 2-11
    combining report formats 1-126
    displaying 1-125
    PDF 1-125

COMPOUND SET parameter 1-125

compressed Millennium VSAM files 8-19

COMPUTE command 4-10

COMT data source 1-151, 2-12

COMT format 1-153

concurrent data source access in Maintain 9-42

configuring deferred server mappings 6-7

configuring internal tables 11-2

configuring NLS (National Language Support) 7-5

configuring special services 6-9

configuring the ReportCaster Distribution Server for NLS 5-11

configuring WebFOCUS Client for NLS 6-20

configuring WebFOCUS client settings 6-10

configuring WebFOCUS Reporting Server for NLS 7-5

configuring WebFOCUS Reporting Server settings 6-4

configuring WebFOCUS startup parameters 6-10

connections 7-4

consolidating financial data 2-62

CONTAINS operator in Maintain 9-35

CONTAINS test 8-40

content blocks 3-14, 3-18

content pages 3-13

Continental Decimal Notation (CDN)
    punctuating large numbers 1-118

controlling missing data 1-43
    in HOLD files 1-43

controlling ORDER fields in indexed lists 8-6

converting file names 1-107

converting totals to Excel 2000 formulas 1-130

cpcodepg.nls file 7-6

cpxcptbl.nls file 7-7

Create Synonym 3-7

CREATE SYNONYM command 7-26, 8-9, 8-24
    for Adabas 8-8
    for Enterprise Java Beans 8-13
    for Essbase 8-14
    restrictions 8-38

creating financial reports 2-56
    for hierarchies 2-56

creating lowercase strings 1-54

creating new columns 4-10

creating temporary fields 4-10

creating uppercase strings 1-63

creating WHERE statements 4-11

CSS in Maintain 9-10

CSSNAME 9-11

CTA files in Maintain 9-12

cumulative normal distribution 1-21

currency symbols 1-121
    extended currency symbols 1-121

custom messages 2-50

Custom Reports 3-8
    creating 3-8
    in Dashboard 3-8, 3-19
    in Managed Reporting 3-8

customizing parsing activity in Maintain 9-8

customizing public views 3-14

customizing the user interface 6-30
    adding .gif files 6-40
    adding text 6-35
    adding text without translation 6-38
    modifying text 6-34

customizing Update Assist designs 2-52

D

Dashboard 3-13
  Add to Favorites option 3-15
  banners 3-19
  domain search in 3-18
  Domain/Role Tree 3-17 to 3-18
  group views 3-13
  public views 3-14
  removing banners 3-14
  toolbars 3-15

Data Adapter for Adabas 8-2

Data Adapter for Allbase 8-9

Data Adapter for DB2 8-10
  connection attributes 8-11
  FETCH FIRST n ROWS clause 8-12

Data Adapter for DBFILE 8-10

Data Adapter for Enterprise Java Beans 8-13

Data Adapter for Essbase 8-14

Data Adapter for Informix 8-19

Data Adapter for J.D. Edwards 8-19

Data Adapter for Micro Focus C-ISAM 8-9

Data Adapter for Millennium 8-19

Data Adapter for MUMPS 8-19

Data Adapter for PeopleSoft 8-20

Data Adapter for Progress 8-20

Data Adapter for SAP BW 8-21

Data Adapter for SAP R/3 8-21

Data Adapter for Siebel 8-24

Data Adapter for UniVerse 8-24

data consolidation 2-63

data sources 1-80
  CENTGL 1-80
  CENTSYSF 1-80

data target options 10-18
  target setup options 10-18, 10-22
  Target Setup tab 10-18, 10-22

data types in Maintain 9-43

Database Management Systems (DBMSs) 8-1

database selection 8-33

DATASET attribute 8-28 to 8-29, 8-31

DATASET syntax for FOCUS data sources 8-30

Date and Time dialog box 4-20

date fields 2-53

date format display 2-8

date formats 2-8
  changing 2-8
  display options 2-8

Date Formats dialog box 2-9

date/time fields 2-51

date/time information in Maintain 9-4

DATEDISPLAY parameter in Maintain procedure 9-20

DB2 CLOB columns 8-39
  adding 8-41
  describing 8-39

DB2 Data Adapter 8-10, 8-13
  table types on OS/400 8-13

DBA files 4-19

DBA in Maintain 9-26, 9-39

DBA prompting in Maintain 2-50

DBAPASS in Maintain 9-39

DBFILE Data Adapter 8-10

COBOL and 8-10
RPG and 8-10

DBMS_ERRORCODE property 9-30, 9-32

DBMSs (Database Management Systems) 8-1

ddname FOCCOMP in Maintain 9-36

ddname MAINTAIN 9-36

ddnames in Maintain 9-36

decimal points 1-111

default data 1-47

DEFCENT parameter in Maintain procedure 9-20

deferred receipts 3-12

DEFINE command 1-106, 4-10
compiling 1-106

DEFINE compiler 1-105

DEFINES parameter 2-69

defining equal ranges 1-27

Deploy Wizard 2-5

deploying a project 2-5, 9-2, 9-4

deploying changed files in Maintain 9-4

descriptors 8-5

designing user interfaces 2-35

Desktop Viewer 2-10

Developer Launch Console 9-6

Developer Studio 2-1 to 2-2
architecture 2-2
Deploy Wizard 2-5
Dimensions Tool 2-6
Explorer 2-4
Managed Reporting for Windows 2-3, 3-3
Project Wizard 2-4

Developer user type 3-2

DEVELOPERS setting 6-12

Dimension Builder 2-6
launching 2-6 to 2-7

Dimensions Tool 2-6 to 2-7

Direct SQL Passthru 8-38

directory structures 6-31
for NLS 6-28

directory tree structure 7-17

display fields 1-98

display formats for reports 1-126
compound reports 1-126
EXL2K 1-133

display options 2-8

displaying children 1-76, 2-56

displaying date formats 2-8

displaying dates 1-131
in EXL2K 1-131

displaying fields 2-70
using functions 2-70

displaying FML hierarchies 2-56

displaying formatted dates 1-131

displaying leading zero 1-119

displaying missing data 1-47

displaying parents and children 1-76, 2-56

displaying reports 1-133
compound 1-125

displaying times 2-9

distribution enhancements 5-6

Domain Admin 3-2

domain searches 3-18

Domain/Role Tree 3-17 to 3-18

drag-and-drop measures 4-2

drill down capability 4-16

drill-down links in Developer Studio, multiple 1-140, 2-21

drill-down links, multiple 1-140

drilling down on dimensions 4-5

drilling down on measures 4-5

DRILLMENUITEM attribute 1-140

DYNAM commands 7-27 to 7-28

dynamic language switch 6-24

dynamic lists 2-49

dynamic parameters 2-46

dynamic reformatting 1-116 to 1-117

dynamic reporting 2-55

dynamic tables of contents 1-142
- adding to reports 1-142
- adding to WebFOCUS StyleSheets 1-142
- for multiple sort groups 1-147

dynamic TOCs (tables of contents) 1-141
- adding to headings 1-145

dynamic tracing 3-20

E

EDAPRINT log 7-16

edaprint.log file 9-16

edastart -console option 7-17

editing language values 6-29

editing the NLS configuration file 7-11

E-Mail Address field 3-2

e-mail notification in ETL 10-19 to 10-20

EMGSRV parameter in Maintain procedure 9-20

Enable control 7-4

enabling Smart Deploy in Maintain 9-4

enclave name in Workload Manager 7-32

enhancing navigation 1-141

Enterprise Java Beans Adapter 8-13

EQ_MASK logical operator in Maintain 9-35

equal ranges 1-27
- defining 1-27

error messages in Update Assist 2-50

ERROROUT command 1-106

Essbase Data Adapter 8-14
- Master File 8-15
- measures dimension 8-15
- missing data 8-17 to 8-18
- non-aggregated fields 8-15
- non-consolidating members 8-16
- SET MEASURE command 8-15
- shared members 8-17
- SUPEMPTY setting 8-18
- SUPMISSING setting 8-17
- SUPSHARE setting 8-17
- SUPZERO setting 8-17
- UDAs (User-Defined Attributes) 8-18
- User-Defined Attributes (UDAs) 8-18
- zero value suppression 8-17
- zero values 8-18

ESSBASE OLAP User-Defined Attributes 8-14

ETL logs 10-4
- managing from the Web Console 10-4

ETL Manager 10-1
- configuring 10-4

ETL request log 10-3
- viewing 10-3

ETL requests 10-2
categories 10-2
logs 10-3
managing from the Web Console 10-4
migrating 10-7 to 10-8, 10-10, 10-12
nesting 10-19
saving as procedures 10-2
scheduling 10-5 to 10-6
sending status by email 10-19 to 10-20

ETL scheduler 10-5
managing from the Web Console 10-5

ETL target tables 10-16

ETL Workbench 10-14

EX command 8-26

examining RDBMS return codes in Maintain 9-32

Excel 2000 1-131
displaying formatted dates 1-131

Excel 97 1-135
reports in 1-135

Excel and Maintain 9-44

Excel reports 1-129, 2-11
selecting languages 1-137

EXEC statements in Maintain 9-38

executing a Java class 8-27

execution enhancements in Maintain 9-4

exit on error 1-106

EXL2K display format 1-133
displaying dates 1-131
formatted dates and 1-131
saving reports 1-130

EXL2K format 1-129, 2-11
FOLD-LINE 1-138

EXL2K FORMULA display format 1-130

EXL2K_DEFCOLW parameter 1-138, 7-11

EXL2KLANG parameter 1-137, 7-11

EXL97 display format 1-135
reports in 1-135

Explorer 9-2
in Developer Studio 2-4

exponential moving average 1-3 to 1-4, 1-9, 2-26 to 2-27, 2-29, 2-31 to 2-32, 2-34

exporting and importing forms in Maintain 9-12

exporting forms in Maintain 9-12

expressions 4-11

extensions in Maintain 9-36

external Cascading Style Sheets (CSS) in Maintain 9-10

extract transform with LAST function 10-15

extract utility 3-12

extracting variable length strings 1-59

F

Favorites 3-15

FDS (FOCUS Database Server) 7-18

FETCH FIRST n ROWS clause 8-12 to 8-13

Field Creator dialog box 4-10

field displays 1-154
adjusting 1-154

field-based reformatting 1-116 to 1-117

Fields tab 4-10

FIELDTYPE = R attribute in Maintain 9-40

FILCASE parameter 1-107

file designations in Maintain 9-36

file names 1-107
converting 1-107

filtering parameter values 2-40

filters 4-11
  for Standard Reports 3-20
  in Dashboard 3-20

financial data 2-62
  consolidating 2-62

Financial Modeling Language (FML) 1-70
  hierarchies 1-70, 1-73
  reports 1-89, 2-64

Financial Report Painter 2-54 to 2-55
  multiple rows 2-69

financial reports 2-55
  charts of accounts 1-76
  formatting 1-89 to 1-90, 2-64
  hierarchies 2-55 to 2-56
  multiple rows 2-69

FML (Financial Modeling Language) 1-70
  hierarchies 1-70, 1-73
  reports 1-89, 2-64

FML (Financial Modeling Language) reports
  formatting 1-89, 2-64

FML hierarchies 1-70, 1-73, 1-76, 2-55
  assigning captions 1-72
  consolidating data 1-78 to 1-79, 1-81
  displaying 1-74
  indenting captions 1-86
  indenting row titles 1-85, 2-56
  indenting text or numbers 1-91

FMLINFO function 1-87 to 1-88

FOCERROR 9-32

FOCHTMLURL parameter 1-105

FOCSBS 7-18

FOCUS data source access in Maintain 9-42

FOCUS Database Server 7-19

FOCUS Database Server (FDS) 7-18

FOLD-LINE clauses 1-138

fonts 1-115
  measuring 1-115

footing limitations 1-98

FOR statement 1-85, 2-69
  repeated values 1-85, 2-69

FORECAST 1-2
  exponential moving average 1-3 to 1-4, 1-9, 2-26 to 2-27, 2-29, 2-31 to 2-32, 2-34
  linear regression analysis 1-3 to 1-4, 2-26 to 2-27, 2-29, 2-31 to 2-32, 2-34
  linear regression equation 1-16 to 1-17
  simple moving average 1-3 to 1-4, 1-6 to 1-7, 2-26 to 2-27, 2-29, 2-31 to 2-32, 2-34

Forecast dialog box 2-27 to 2-34

forecasting 1-2, 1-18, 2-26

FORMAT EXL97 command 1-135

formatted dates 1-131
  Excel 2000 and 1-131

formatting cells 1-89 to 1-90, 2-64, 2-68

formatting columns 1-89, 2-64 to 2-65

formatting options 1-89
  financial reports 1-89, 2-64

formatting rows 1-89, 2-64, 2-67

forms 2-46
  running from Maintain 9-6
  supplying parameter values 2-38 to 2-39

forms and Cascading Style Sheets 9-10

forms in Maintain 9-12

formulas 1-130
  converting to Excel 2000 1-130

FORMULTIPLE parameter 2-69

Frame Properties window 2-47

frame separator display 3-18

frames 2-47
  adding to HTML pages 2-47
  Frame Properties window 2-47

free text 1-89
  formatting 1-89, 2-64

FTP parameters 10-19

functions 1-52, 1-155 to 1-156, 2-70
  and AnV fields 1-52
  LENV 1-53
  LOCASV 1-53 to 1-54
  NORMSDST 1-21
  NORMSINV 1-21
  POSITV 1-56
  SQUEEZ 1-155, 2-70
  STRIP 1-156, 2-70
  SUBSTV 1-59
  TRIM 1-158, 2-70
  TRIMV 1-61
  UPCASV 1-63
  validating arguments 1-66

G

generating formulas in Excel 2000 1-133

GET CHILDREN command 1-73, 1-78

GET CHILDREN parameter 2-56

GetAppCGIValue Maintain function 9-33

GFA content 9-12

GFA files 9-4
  deployment information 9-4

GFBID (Global Format Buffer ID) 8-7

Global Format Buffer ID (GFBID) 8-7

global settings 6-15

graph enhancements 1-93
  printing output 1-97
  type options 1-97

Graph Preview 4-18

graph styles 4-18

graph types 4-18

GRAPHSERVURL parameter 2-69

GRAPHSERVURL SET parameter 1-93 to 1-95

group views 3-13

grouping 1-25
  numeric data 1-25
  sort fields 1-27

grouping numeric data 1-27
  into tiles 1-25 to 1-26, 4-6

grouping numeric data in tiles 4-7

GTREND SET parameter 1-97

H

HDAY parameter 1-104, 2-69

HDAY SET parameter 1-104

heading limitations 1-98

Help option 5-10

hidden reports 4-5

hiding content blocks 3-18

hierarchies 2-56
  displaying 2-56
  displaying values as captions 2-56 to 2-57
  indenting levels 2-56

Hiper Memory 7-28

HNODATA command 1-43 to 1-44

HOLD command 1-152

HOLD formats 1-48, 1-154
  COM 1-152
  COMT 1-153
  TAB 1-153
  TABT 1-153

HOLDMISS command 1-47

HOLDMISS parameter 1-47

holidays 1-104
specifying 1-104

horizontal sort field values 1-149

HTML files 6-33

HTML Graph Assistant 4-17
enhancements 4-17

HTML Report Assistant 4-8
creating new columns 4-10
creating temporary fields 4-10
enhancements 4-19
redesign 4-9
selecting records for a report 4-11

HTML reports 1-141

HTML tables of contents 1-142
adding to reports 1-142
adding to WebFOCUS StyleSheets 1-142
for multiple sort groups 1-147

HTML TOCs (tables of contents) 1-141
adding to headings 1-145

HTTP basic authentication 7-24

hyper descriptors 8-5

I

ibi_cgi in Maintain 9-3

ibi_start in Maintain 9-3

ibimultilanguage.js file 6-29

IBISTART.WFS file in Maintain 9-39

IBIWF_language function call 6-27

IDENTIFY columns 9-40

identifying cells 1-89, 2-64

identifying columns 1-89, 2-64 to 2-65

identifying rows 1-89, 2-64, 2-67

IDoc listener 8-23

IF screening test 8-25

Image Properties window 2-43

images 2-43
adding to HTML pages 2-43
Image Properties window 2-43

importing and binding variables in Maintain 9-18

importing forms in Maintain 9-12

importing variables in Maintain 9-17

importing WebFOCUS parameters in Maintain 9-19

indenting hierarchy levels 2-56

Informix Data Adapter 8-19

IN-GROUPS-OF phrase 1-27

IN-RANGES-OF phrase 1-27 to 1-28

inserting 1-29
page numbers 1-29

INT2CHAR Maintain function 9-33

integration of Maintain and Developer Studio 9-2

Intermediate Documents (IDocs) 8-23

Internal Sequence Number (ISN) 8-6, 8-8

internal tables 11-2

INTERNAL type security settings 7-31

inverted lists for Adabas 8-5

ISN (Internal Sequence Number) 8-6, 8-8

issuing SQL commands in Maintain 9-30

iWay Web Console 10-4
configuring ETL Manager 10-4

IWC.GetAppCGIValue function 9-16

J

J.D. Edwards Data Adapter 8-19

Java Applications Adapter 8-26

Java classes 8-26 to 8-27
- CALLJAVA command 8-26
- communication 8-27
- EX command 8-26
- writing 8-27

Java programs 8-27

JavaScript 4-2

JavaScript in Maintain 9-13, 9-37

job ID 10-4

join structures 4-15

joins 8-38
- between data sources 8-38
- between DBMSs 8-38

JSURLS parameter 1-104

JUSTIFY attribute 1-115

K

keywords 7-2

L

LABEL attribute 1-89 to 1-90, 2-64

labels 1-89
- formatting cells and 1-90
- formatting rows and 1-89, 2-64

LANG parameter 1-161

LANGUAGE parameter in Maintain procedure 9-20

language subdirectories 6-31

language values 7-10

language values for OS/400 7-13

language values for UNIX 7-12

language values for Windows 7-12

Large Volume Requests report 11-7 to 11-8

LAST function 10-15

launch pages in Maintain 9-14

launching Dimension Builder 2-6 to 2-7

leading zero 1-119
- displaying 1-119

LENV function 1-53

limitations for headings and footings 1-98

linear regression 1-97

linear regression analysis 1-3 to 1-4, 2-26 to 2-27, 2-29, 2-31 to 2-32, 2-34

linear regression equation 1-16 to 1-17

List mode 4-10

listener statistics 7-33

load options 10-17

LOCASV function 1-53 to 1-54

log continuation files 7-16

log file (ETL) 10-3

Log File option 5-10

Log tables 10-4

Logoff option 5-10

long procedure names 11-6

Long-running Requests report 11-7

LOOKGRAPH command 1-97

LOOKGRAPH parameter 1-97

M

Maintain and Developer Studio integration 9-2

Maintain and reporting integration 9-3

Managed Reporting 3-6
- Change Management 3-4
- creating synonyms 3-7
- Developer for Windows 3-3
- migrating 3-12
- multiple servers 3-7

Managed Reporting Developer for Windows 2-3

managing the visible property in Maintain 9-13

Master Files 1-124, 1-152
- editing in Maintain 9-8
- for CSV format 1-152
- for Essbase 8-15
- Minus Edit format option 1-124

max_connections_per_user keyword 7-2

maxFolders variable 2-51

maximum display fields 1-98

maxNodesPerFolder variable 2-51

MEASURE setting 8-15

measures dimension for Essbase 8-15

measuring for column width alignment 1-115

measuring for decimal alignment 1-115

memory enhancements in Maintain 9-4

MESSAGE parameter in Maintain procedure 9-20

metadata generation 8-32

Micro Focus C-ISAM Data Adapter 8-9

Microsoft Access and Maintain 9-43

migrating ETL requests 10-7

migrating ETL requests on OS/390 10-10

migrating ETL requests on OS/400 10-12

migrating ETL requests on UNIX and Windows 10-8

Migration Utility 3-12

Millennium Data Adapter 8-19

mime.wfs configuration settings 6-18

Minus Edit format option 1-124

minus signs 1-124
- displaying 1-124

missing data characters 1-43 to 1-47

missing data suppression in Essbase 8-17

MNTPERF trace 9-15

MNTSTMT Master File in Maintain 9-15

MNTSTMT trace 9-15

modifying server settings 6-4

MQ XML listener 7-33
- configuring 7-33

multiLanguage variable 6-29

multi-level table of contents 1-141

multiple content pages 3-13

multiple drill-down links 1-140

multiple drill-down links in Developer Studio 1-140, 2-21

multiple servers 3-7

multiple spaces in fields 1-155

multiple-level tracing in Maintain 9-15

multivariate regressions 1-18
- and calculating trends 1-18
- and predicting values 1-18

MUMPS Data Adapter 8-19

MVS external sort 7-28

MVS type security settings 7-31

MVSAPP feature 7-29

N

name spaces 10-2

National Language Support (NLS) 6-20, 7-5
  configuring a WebFOCUS Client 6-20
  configuring a WebFOCUS Reporting Server 7-9

native Excel formulas 1-133

Native Interface 8-21
  support 8-23

navbar.htm file 2-51

navigation 1-141

NE_MASK logical operator in Maintain 9-35

nesting requests in ETL 10-19

new report columns 4-10

NEXT command in Maintain and screening options 9-35

NLS (National Language Support) 6-20, 6-23
  configuring a WebFOCUS Client 6-20
  configuring a WebFOCUS Reporting Server 6-20
  directory structures 6-28
  dynamic language switch 6-24
  supported languages 6-28

NLS configuration file 7-11

NLS Configuration Wizard 7-5, 7-9

nlscfg.err file 6-21, 7-11

NLSCHR Maintain function 9-33

NOBREAK phrase 1-125

NODATA parameter in Maintain procedure 9-20

noibitranslate flag 6-33, 6-38, 6-40

non-aggregated fields for Essbase 8-15

non-consolidating members in Essbase 8-16

None option 4-16

non-FOCUS files 7-32

non-standard aliases 1-105
  establishing 1-105

normal distribution 1-21

NORMSDIST function 1-23 to 1-24

NORMSDST function 1-21

NORMSINV function 1-21, 1-23 to 1-24

NOT_IN logical operator in Maintain 9-35

numbers 1-22
  standard normal deviation 1-22

numeric data 1-25, 1-27, 4-6

numeric data grouped in tiles 4-7

numeric functions 1-23
  FMLINFO 1-87 to 1-88
  NORMSDIST 1-23 to 1-24
  NORMSINV 1-23 to 1-24

O

ODBC connection 8-20

OLAP 4-2
  drag-and-drop dimensions in 4-2
  grouping data with tiles 4-6
  hidden reports in 4-5
  right-click options 4-3
  tile fields 4-6

OLAP options 4-16

OLAP tools 4-5
  selecting 4-5

OLAP-enabling reports 4-5

OMITS operator in Maintain 9-35

OMITS test 8-40

OpenVMS Operating System 7-32

optimized loading in ETL 10-21

options settings 3-19

ORDER field for Adabas 8-6

Ordinary Least Squares 1-18, 2-26

OS/400 language and code page values 7-13

OS/400 Operating System 7-32

output file formats 1-152
- COM 1-152
- COMT 1-153
- TAB 1-153
- TABT 1-154

output formats 1-151, 2-12

P

parameter values 2-36, 2-38 to 2-40
- dynamic list 2-46
- filtering with the chain feature 2-40
- static list 2-36

parameters 2-40
- Parameters window 2-38 to 2-40, 2-46
- properties 2-38 to 2-40

Parameters window 2-38 to 2-40, 2-46

parentheses 4-11
- grouping expressions together 4-11
- grouping WHERE Statements together 4-11

parsing in Maintain 9-8, 9-37

PASS parameter in Maintain procedure 9-20

passing language parameters to a URL 6-27

passwords in Maintain 9-39

PATH keyword in Maintain 9-38

PCHOLD command 1-142
- Excel formats and 1-135

PCOMMA parameter 2-69

PDF reports 1-160
- adding PostScript fonts 1-160

PeopleSoft Data Adapter 8-20

Percent Edit Format Option 1-124, 2-10

percentiles 4-6 to 4-7

performance enhancements in Maintain 9-4

phonetic descriptors 8-5

POSITV function 1-56

PostScript fonts 1-160

predicting values 1-2
- and multivariate regression 1-18
- FORECAST 1-3, 2-26

prefix operators 1-31
- and calculated values 1-38
- and multiple commands 1-40
- on summary values 1-32 to 1-33

preselecting items by value in Maintain 9-13

Preview screen 4-18

printing graph output 1-97

procedures 9-6
- running from Maintain 9-6

procedures in WebFOCUS Reporting Server Console 9-2

profile keyword 7-2

profiles directory 7-4

Progress Data Adapter 8-20

project search in Maintain 9-9

Project Wizard 2-4

projects and applications 9-2

public users 3-18

public views 3-13 to 3-14

punctuating large numbers 1-118 to 1-119

Q

quartiles 4-6 to 4-7

queued connections 7-4

Quick Links 6-22

Quiesce control 7-4

-quit option 7-17

R

radio buttons 2-44
    Radio Buttons Properties window 2-45

Radio Buttons Properties window 2-45

ranges 1-27 to 1-28

RDBMS return codes in Maintain 9-32

READLIMIT tests 8-12

Recent link 3-15

RECOMPUTE command 1-31
    prefix operators and 1-32 to 1-33

records 4-11
    selecting 4-11

recreating synonyms 2-7

redirection settings 6-18

reducing column width 1-138

reformatting fields 1-116 to 1-117

REGRESS 1-2
    and multivariate regression 1-18

REGRESS method 1-18, 2-26

regression equations 1-18

Remote Procedure Calls (RPCs) 7-28

remote server control 7-2

remote server installation 7-2

Remove All button 3-15

removing banners 3-14

removing characters from fields 1-156

removing characters from strings 1-61

removing parameters from a chain 2-41

removing punctuation 1-156

Report Assistant 4-5
    OLAP enabling from 4-5

Report Assistant OLAP settings 4-5

report fields 4-9

Report Library 3-2, 5-9
    in Dashboard 3-20

Report Painter 2-14
    formatting capabilities 2-14 to 2-16
    OLAP enabling from 4-5

report types 3-8
    custom 3-8

ReportCaster 3-2, 5-2
    architecture 5-2
    configuration 5-2
    Development and Administration Interface 5-5
    HTML user interface 5-8
    in Dashboard 3-20

ReportCaster Console 5-10
    Help option 5-10
    Log File option 5-10
    Logoff option 5-10
    Run Now option 5-10
    Users option 5-10

ReportCaster Distribution Server 5-11
    configuring for NLS 5-11
    settings 5-13

reporting against hierarchies 1-76, 2-55 to 2-56

reporting enhancements 1-124

Reporting Object components 3-7

reports 1-29
- Candidates for Column Index 11-9
- compound 1-125
- expanded limits 1-98
- including script files 1-104
- inserting page numbers 1-29
- Large Volume Requests 11-8
- Long-running Requests 11-7
- OLAP-enabling 4-5
- saving 1-130

repository extract utility 3-12

request logs 10-3

Resource Analyzer 11-1
- accessing through Web console 11-6
- Candidates for Column Index report 11-7
- configuring internal tables 11-2
- Large Volume Requests report 11-7
- long procedure names 11-6
- Long-running Requests report 11-7

Resource Analyzer reports 11-9
- Large Volume Request reports 11-8
- Long-running Requests report 11-7

resource files 6-30
- interpreting 6-33
- locating names in 6-32

Resource Governor 11-1
- accessing through Web console 11-6
- configuring internal tables 11-2
- long procedure names 11-6

Resource Layout Painter 2-35, 2-43
- adding frames 2-47
- adding images 2-43

RESTRICTSUM command 8-14

retaining SUBSTRING behavior 8-42

retrieving CGI parameters from Maintain

procedures 9-16

retrieving records 8-25

return codes in Maintain 9-32

reusing form templates in Maintain 9-12

reversing the order of parameters 2-41

right-click menu options in Update applications 2-53

right-click options 4-3

row totals 1-149

rows 1-89
- formatting 1-89, 2-64, 2-67
- identifying 1-89, 2-64, 2-67
- identifying in WebFOCUS StyleSheets 1-89, 2-64

RPCs (Remote Procedure Calls) 7-28

Run Now option 5-10

Run Procedure in Maintain 9-37

S

SAP BW Data Adapter 8-21

SAP processing 8-23

SAP R/3 Data Adapter 8-21

SAP R/3 Interface 8-22
- Native Interface enhancement 8-22

SAP R/3 Native Interface 8-21 to 8-23
- BAPI support for 8-22

SAS (Shared Application Server) in Maintain 9-42

SAVE formats 1-152
- COM 1-152
- COMT 1-153
- TAB 1-153
- TABT 1-153

saving graphs as GIF files 1-95
    using SET GRAPHSERVURL 1-93
saving report output 1-130
scatter graph 1-97
scheduling ETL requests 10-5
    Scheduler tool 10-5 to 10-6
scheduling wizard 5-8
screening options for NEXT command 9-35
search order in Maintain 9-36
search paths in Maintain 9-38
searching a project in Maintain 9-9
searching for text in Maintain 9-9
Secure Socket Layer (SSL) 7-24
security 7-21
    administration levels 7-22
    administrator privileges 7-22
    methods 7-21
    on Windows NT/2000 7-21
    passwords 7-23
    specifying Windows NT/2000 clients 7-23
    SSL (Secure Socket Layer) 7-24
    TLS (Transport Layer Security) 7-24
    user privileges 7-22
    user rights 7-21
    WCPROTECT mode 7-23
security settings 7-31
    INTERNAL type 7-31
    MVS type 7-31
SelectedItems property for Maintain controls 9-13
selecting code pages 7-6
selecting databases from Web Console 8-33
selecting OLAP tools 4-5
selecting records 4-11
selecting records in VSAM data sources 8-25
selecting report fields 4-10
Selection Criteria tab 4-11
server activity file 7-16
server configuration file 7-23
server control 7-2
Server for OS/390 and z/OS 7-26
    accessing synonyms from 7-26
    CREATE SYNONYM command 7-26
    DYNAM command 7-27
    MVS external sort 7-28
    security settings 7-31
    supported data adapters 7-26
server installation 7-2
server level profiles 7-4
server nodes 6-8
    alternate 6-8
server search behavior 3-7
servers 7-2, 7-4
    configuring 7-4
    Enable control 7-4
    managing connections to 7-4
    parameters for 7-2
    Quiesce control 7-4
    user level profiles 7-4
SET AGGREGATE NOOP command 8-14, 8-16
SET BLANKINDENT command 1-85, 2-56, 2-69
SET commands 1-142, 1-161, 2-69
    ERROROUT 1-106
    HOLDMISS 1-47
    PSPAGESETUP 1-161
    SUMMARYLINES 1-37
SET EMPTY command 8-14
SET FORMULTIPLE parameter 2-69
SET HNODATA command 1-43

SET MEASURE command 8-14 to 8-15

SET NOPACCESS command 8-8 to 8-9

SET ORDERKEYS command 8-6

SET parameters 1-162, 2-69
- BLANKINDENT 1-85, 2-69
- CDN 1-118, 2-69
- CENT-ZERO 1-119, 2-69
- changing from Maintain procedure 9-20
- COMPOUND 1-125
- DEFINES 2-69
- EXL2KLANG 1-137
- FILCASE 1-107
- FOCHTMLURL 1-105
- FORMULTIPLE 2-69
- GRAPHSERVURL 1-93 to 1-95, 2-69
- GTREND 1-97
- HDAY 1-104, 2-69
- HNODATA 1-43
- JSURLS 1-104
- LANG 1-161
- LANG parameter 1-161
- LOOKKGRAPH 1-97
- PCOMMA 2-69

SET PSPAGESETUP command 1-161

SET RESTICTNUM command 8-14

SET RESTRICTNUM command 8-16

SET SUMMARYLINES command 1-37

SET SUPEMPTY command 8-18

SET SUPMISSING command 8-14, 8-17

SET SUPSHARE command 8-14, 8-17

SET SUPZEROS command 8-14, 8-17

SET SYNONYM command 8-4

SET USERFCHK command 1-66

SET USERFNS command 1-67

setting CDN from Maintain procedure 9-22

setting COMMIT from Maintain procedure 9-22

setting DATEDISPLAY from Maintain procedure 9-23

setting DEFCENT from Maintain procedure 9-23

setting EMGSRV from Maintain procedure 9-24

setting LANGUAGE from Maintain procedure 9-25

setting MESSAGE from Maintain procedure 9-25

setting NODATA from Maintain procedure 9-26

setting PASS from Maintain procedure 9-26

setting statement trace in Maintain 9-15

setting TRACEOFF from Maintain procedure 9-27

setting TRACEON from Maintain procedure 9-27

setting TRACEUSER from Maintain procedure 9-28

setting up Shared Application Server in Maintain 9-43

setting USER from Maintain procedure 9-26, 9-28

setting WARNING from Maintain procedure 9-28

setting YRTHRESH from Maintain procedure 9-29

Shared Application Server (SAS) in Maintain 9-42

shared Custom Reports 3-11

Shared Library Path 8-35

shared members for Essbase 8-17

Siebel Data Adapter 8-24

simple moving average 1-3 to 1-4, 1-6 to 1-7, 2-26 to 2-27, 2-29, 2-31 to 2-32, 2-34

simultaneous use in Maintain 9-42

Smart Deploy in Maintain 9-4

SMF records 7-31

SOAP/XML listener 7-33

sort fields 1-142
defining ranges 1-27
grouping 1-147
sort footings 1-98
limitations 1-98
sort headings 1-98
limitations 1-98
sort order 1-27
grouping numeric data 4-7
in numeric data 1-27
sort values 1-27
sorting tables 2-54
source control 2-48
source file management 2-48
Source object 10-15
specifying code pages manually 7-10
specifying holidays 1-104
specifying indentation between levels 1-85, 2-56
specifying language values manually 7-10
specifying search paths in Maintain 9-38
SQL commands in Maintain 9-30
SQL EDA commands in Maintain 9-30
SQL SELECT statements 10-15
SQL/Answer set object 10-15
SQUEEZ function 1-155 to 1-156, 2-70
SSL (Secure Socket Layer) 7-24
stacking columns 1-138
in EXL2K format 1-138
standard normal deviation 1-22
Standard Reports 3-11, 3-20
statement trace in Maintain 9-15
Statistics tables 10-4
STRIP function 1-156 to 1-157, 2-70
style sheets 1-145
WebFOCUS StyleSheets 1-145
Styles tab 4-18
STYLESHEET parameter 1-145
TOC 1-145
SUBSTRING function 8-42
SUBSTV function 1-59
SUB-TOTAL 1-40
using multiple commands 1-40
SUB-TOTAL command 1-31
prefix operators and 1-32 to 1-33
SUBTOTAL command 1-31
prefix operators and 1-32 to 1-33
SUFFIX values 1-151, 2-12
SUMMARIZE 1-40
using multiple commands 1-40
SUMMARIZE command 1-31
prefix operators and 1-32 to 1-33
summary line processing 1-37
summary values 1-31
prefix operators and 1-32 to 1-33
SUPEMPTY setting for Essbase 8-18
superdescriptors for Adabas 8-4
SUPMISSING setting for Essbase 8-17
supported code pages 7-7
supported languages 6-28
suppressing missing data in Essbase 8-17
suppressing zero values in Essbase 8-17
SUPSHARE setting for Essbase 8-17

SUPZERO setting for Essbase 8-17

switching access modes 8-8 to 8-9

switching between languages 6-24

Synonym Wizard 2-7
    recreating synonyms 2-7

synonyms 2-7, 3-7, 7-26
    accessing from OS/390 and z/OS 7-26
    creating through Server Console 7-26
    recreating 2-7

syntax parsing in Maintain 9-37

SYS_MGR.DBMS_ERRORCODE 9-32

SYS_MGR.ENGINE 9-30

SYS_MGR.FOCSET 9-20

T

TAB format 1-153

tab names 1-134

tab-delimited data sources 2-12

tab-delimited files 1-48
    extract files 1-154
    output files 1-153

TABLASTPAGE variable 1-29

Table of Contents (TOC) 1-141

table types on OS/400 8-13

TABT data source 1-151, 2-12

TABT format 1-153 to 1-154

target object 10-16

Target Setup tab 10-18
    FOCUS files 10-18
    Fusion files 10-18
    relational data sources 10-18, 10-22

target tables in ETL 10-16
    pre-loading options 10-17

tasks 5-6
    distributing 5-6
    file type 5-6
    report type 5-6
    URL type 5-6

temporary columns 10-15
    creating 10-15

temporary fields 4-10

testing procedures in Maintain 9-37

Text Console 7-17

text searching in Maintain 9-9

TILE column 1-25, 4-6 to 4-7

tile fields 1-25, 4-6 to 4-7

TILES parameter 1-25

TILES phrase 4-6

time display 2-8

time display options 2-9

time formats 2-9

timestamp columns 9-40

TLS (Transport Layer Security) 7-24

TOC (Table of Contents) 1-141

toolbars 3-15
    in Dashboard 3-15, 3-17

toolboxes 3-16

trace facility in Maintain 9-15

TRACEOFF parameter in Maintain procedure 9-20

TRACEON parameter in Maintain procedure 9-20

TRACEUSER parameter in Maintain procedure 9-20

tracing for the WebFOCUS Client 6-22

tracing in Dashboard 3-20

tracing in Maintain 9-16

transcoding tables 7-5

translating formulas to Excel 2000 1-130

transparent superdescriptors 8-4

Transport Layer Security (TLS) 7-24

transport object in ETL 10-19

transport options 10-19
    FTP 10-19
    Transport window 10-19

Transport window 10-19

tree navigation control in Update Assist 2-51

trends in data 1-2, 1-18, 2-26

TRIM function 1-158 to 1-159, 2-70

TRIMV function 1-61

Tru64 operating systems security 7-25

TX fields 8-39 to 8-40, 8-42
    describing 8-39
    relational data adapters and 8-39

TYPE ON EDAPRINT in Maintain 9-16

U

UDAs (User-Defined Attributes) for Essbase 8-18

UniVerse Data Adapter 8-24

UNIX language and code page values 7-12

UNIX Operating System 7-32

UPCASV function 1-63

Update Assist tool 2-49, 2-51
    color schemes for applications 2-52
    controlling date fields 2-53
    customizing designs 2-52
    right-click menu options 2-53
    sorting tables 2-54
    validation option 2-51

updating applications 2-49

updating graphs 4-19

URLs 6-27
    drill-downs within 5-6
    passing language parameters 6-27

user administration 3-2

user interfaces 2-35

user level profiles 7-4

USER parameter in Maintain procedure 9-20

user rights 7-21

User-Defined Attributes (UDAs) for Essbase 8-18

USERFCHK setting 1-66

USERFNS setting 1-67

Users option 5-10

V

validating function arguments 1-66

validation option in Update Assist tool 2-51

VARCHAR 1-50

varchar data type 9-43

VARCHAR support 8-42

variable varchar 9-43

variable-length expressions 1-50

variables, binding in Maintain 9-17

variables, importing in Maintain 9-17

VBScript in Maintain 9-13, 9-37

viewing reports 2-10

virtual directories in Maintain 9-3

visible property for Maintain controls 9-13

VSAM data sources 8-25
    selecting records and 8-25

W

WARNING parameter in Maintain procedure 9-20

WCPROTECT mode 7-23

Web aliases in Maintain 9-3

WebFOCUS Browser Version 4-5
    OLAP and 4-5

WebFOCUS Client
    configuring for NLS 6-20

WebFOCUS Client Console 6-2

WebFOCUS Client settings 6-15
    customizing 6-20

WebFOCUS Client tracing 6-22

WebFOCUS Connector for Excel 9-44

WebFOCUS Developer Studio 4-5
    OLAP and 4-5

WebFOCUS Reporting Server Console 7-5
    Maintain procedures 9-2
    Maintain trace output 9-15

WebFOCUS StyleSheets 1-145

wfmpriv1.htm launch page 9-14

wfmstart.html launch page 9-14

WHERE clause 8-39 to 8-40

WHERE clause in NEXT commands 9-35

WHERE statements 4-11

WHERE TOTAL statements 4-11

WIDTH attribute
    measuring 1-115

Windows language and code page values 7-12

WITH CHILDREN command 1-73, 1-78

WITH STATISTICS option 8-38

WLM (Workload Manager) 7-32

workflow tool 10-19
    Transport component 10-19

Workload Manager (WLM) 7-32

worksheet customization 1-129, 2-11

worksheet names 1-134

X

X-axis values 1-97

XMI Client/Server Data Adapter 7-29 to 7-30

XMI listener 7-29 to 7-30

Y

Y-axis values 1-97

YRTHRESH parameter in Maintain procedure 9-20

Z

zero values 8-18
    suppressing in Essbase 8-18

Reader Comments

In an ongoing effort to produce effective documentation, the Documentation Services staff at Information Builders welcomes any opinion you can offer regarding this manual.

Please use this form to relay suggestions for improving this publication or to alert us to corrections. Identify specific pages where applicable. You can contact us through the following methods:

Mail: Documentation Services - Customer Support
Information Builders, Inc.
Two Penn Plaza
New York, NY 10121-2898

Fax: (212) 967-0460

E-mail: books_info@ibi.com

Web form: http://www.informationbuilders.com/bookstore/derf.html

Name:_____

Company:_____

Address:_____

Telephone:_____Date:_____

E-mail:_____

Comments:

Reader Comments

Information Builders, Two Penn Plaza, New York, NY 10121-2898 **(212) 736-4433**